Buyer's Guide To The Latest Printers

COMPUTE!

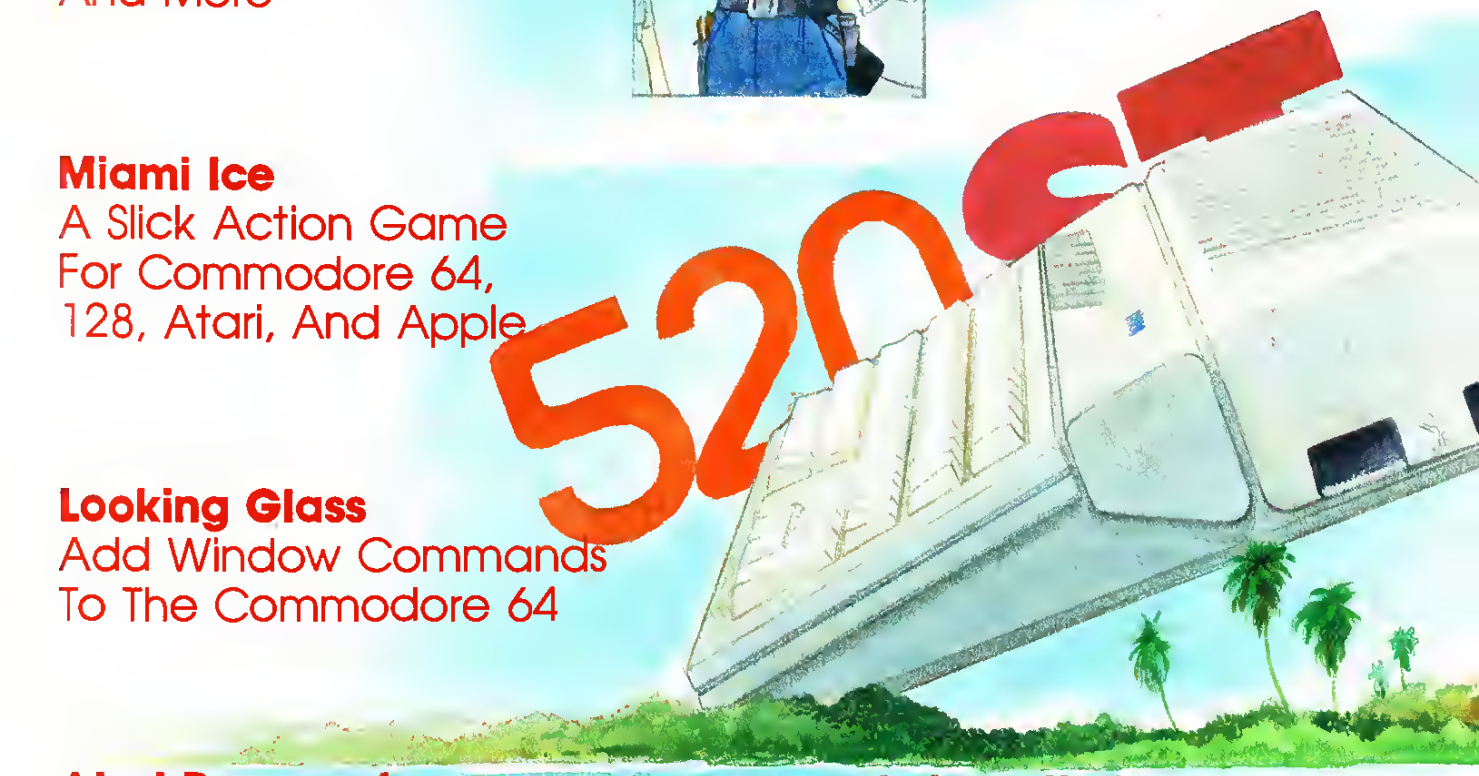
$3.00
June
1986
Issue 73
Vol. 8, No. 6
$3.95 Canada
02193
ISSN 0194-357X

The Leading Magazine Of Home, Educational, And Recreational Computing

Atari ST Hints & Tips
How To Customize Icons, Autorun Programs, And More

Miami Ice
A Slick Action Game For Commodore 64, 128, Atari, And Apple

Looking Glass
Add Window Commands To The Commodore 64



Atari Password
Protect Your Programs From Prying Eyes

Automatic Typist
Using Exec Files On The Apple II

Meet ED:
The AmigaDOS Text Editor

Minding Memory From IBM BASIC
Put An End To RAM Wars

71486 02193


MichTron
DIGITAL RESEARCH
BATTERIES INCLUDED
EPYX
MIGRAPH
SIERRA
ACTIVISION
subLOGIC
SPINNAKER
and in walks Della, smiling mysteriously. "Looks to me like you had yourself a female visitor last night, Perry. Entertaining a new client?"
INFOCOM
You have:
no tea
a loose pile of junk mail
a toothbrush
a flathead screwdriver
your gown (being worn)
It looks like your gown contains:
pocket fluff
The vast yellow ships thunder across the sky, spreading waves of terror and panic in their wake. The voice of the Vogon Captain slams across the country, insisting that the planning charts and demolition orders have been available at the local planning office in Alpha Centauri for fifty years and it's too late to

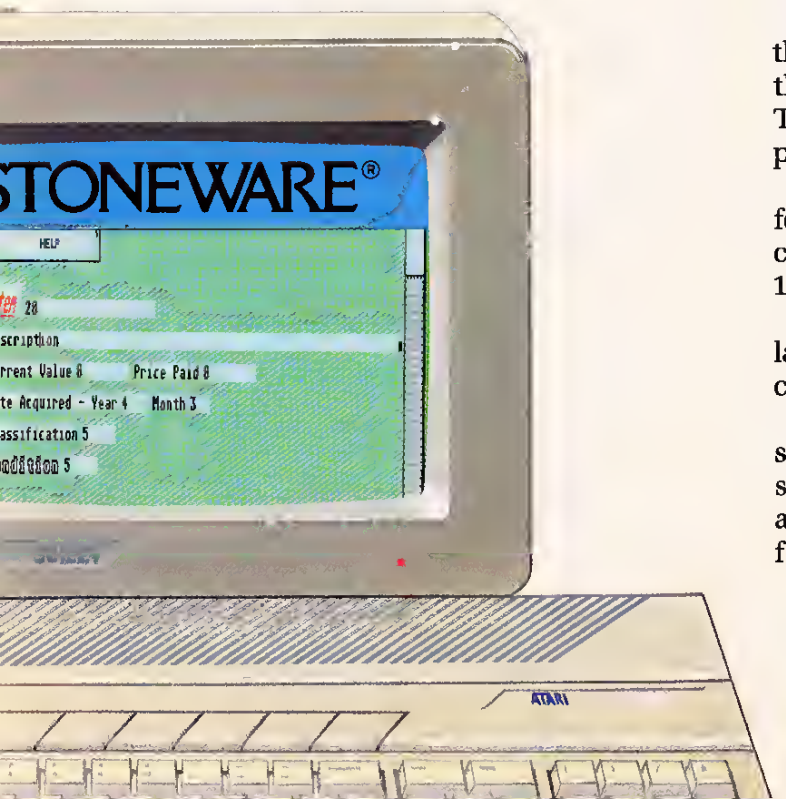
THE ST COMPUTER LINE FROM ATARI.

IT'S ALREADY KNOWN BY THE COMPANY IT KEEPS.

It's only natural that the hottest new computers in America are attracting the hottest software companies in the business.

The library of innovative business, education, entertainment, system management, and integrated package software for the 520ST™ and the new 1040ST™ is already impressive, with literally dozens of new programs being introduced almost every week.

In fact, the software companies who are committing their time, money, and expertise to the ST are the same companies who regularly show up on all the software hit lists.



And it's no wonder that the leading software developers are excited by the power and speed of the ST Computers.

Stoneware®, for example, checked out the speed of the ST Disk Drive in data base applications and flipped. Instead of having to wait forever to manipulate data, thousands of records can be sorted in a fraction of the time that it takes on other computers. And instant responsiveness is the name of the game, not waiting.

Sierra On-Line®, on the other hand, took one look at our incredible high speed, high resolution graphics and was ecstatic. The result is a whole series of games that are more realistic and lifelike than ever before.

For their needs, Activision® focused on the built-in MIDI ports for attaching synthesizers and other musical instruments. This enabled them to design the ultimate program for playing and composing music.

The list goes on and on. But a designer for Spinnaker® perhaps summed up the capabilities of the 520ST and the new 1040ST best:

"I feel like a painter," he said, "who at last has a canvas large enough to let my creativity run free."

So if you're looking for a computer system that combines the very best in software with the very best in hardware at the very best price . . . you're looking for an ST from Atari.

ATARI®

Thomson offers a whole new world of graphics capabilities for your Commodore or Apple IIC or IIE computers. And Thomson offers you more than the competition.

Thomson monitors offer these advantages:

- *Compatible with IBM,® Apple,® Commodore,® Atari™ and others*
- *The choice of monochrome or color monitors with TV-grade to high resolution text and graphics*
- *Monochrome text-editing mode for color monitors*
- *Alternate use as cable or VCR monitor*
- *Broad range of the features you need at the prices you want*

Because of its versatility and compatibility, you can still use your Thomson monitor if you switch computers; you'll never outgrow the capabilities of a Thomson monitor.

Who is Thomson? Thomson is a six billion dollar multi-national company. Unlike our competition, we design and manufacture our own monitors; so the quality you get is constant and superb.

Your Apple or Commodore computer is a great body. Choose a Thomson monitor, and give it a beautiful face.

For the name of the Thomson dealer nearest you, call 1-800-325-0464. In California call 1-213-568-1002. (Monday-Friday, 9 a.m. to 5:00 p.m. PST)

THOMSON

A SIGHT FOR SORE EYES.™

 5731 W Slauson Avenue, Suite 111, Culver City, CA 90230

HOW DO YOU TOP APPLE, COMMODORE, OR IBM BEAUTIFULLY?

SEE YOUR THOMSON DEALER.

Alabama
Mel's Photo Shop, Montrose
Mall Ted, Mobile
Conrad Computer, Birmingham
Command Computer, Birmingham

Alaska
Fred Meyer Photo Electronic Centers:
Juneau, Anchorage, Fairbanks
Hy-Tech Systems, Juneau

Arizona
Compu Save, Phoenix
National Business Software, Tempe

California
Bill's TV, Santa Clara
P.C. Edge, Santa Clara
Grand Central Software, Watsonville
Los Altos PC, Los Altos
Davis Office Systems, Merced
Computer Biz: San Jose, Cupertino
Computer Nook, San Bernardino
Personal Computer Service,
Marina del Rey

Colorado
Software Etc., Denver
Computer Dynamics, Colorado Springs
Computer Interface, Golden
MSR Inc., Denver
Classic Computers, Fort Collins
CRT, Fort Collins
Computer Source, Boulder
Alpha Center, Denver

Connecticut
Crazy Eddie's, All Locations

Florida
Random Access, Ft. Walton Beach
Office World, Sarasota
AA Computer, Jacksonville
Computer Base, Tampa
Discount Disc, Shalimar
G.C.C.E., Panama

Hawaii
Computer House, Honolulu
PC Price Busters, Honolulu

Idaho
Fred Meyer Photo Electronic Centers:
Boise, Pocatello, Idaho Falls

Illinois
Aardvark Computer, Chicago Heights
Protecto, Barrington

Kansas
Computer Center Inc., Olathe

Minnesota
Computer Exercise World, Minneapolis

Mississippi
Enterprise Unlimited, Jackson

Missouri
Micronomics, Winchester

Nevada
Computer House, Sparks

New Jersey
Computer Software Sales, Ocean
Super Software, E. Brunswick
Crazy Eddie's, All Locations

New Mexico
Suntec Computers, Albuquerque
Omega Business Products, Albuquerque

New York
Sysut Computers, Brooklyn
Crazy Eddie's, All Locations

North Carolina
Triad Computer, Greensboro
Hamilton Computer, Greensboro
Chips Computer, Fayetteville
Two M. Systems, Jacksonville

North Dakota
Computerland, Fargo

Ohio
Microwave Magic, Fairfield
Electronic Connection, Centerville
Fairborn Home Computers, Fairborn
Quality Computer Application, Toledo

Oklahoma
Master Systems Ltd., Bethany

Oregon
Schad's Electronics, Eugene
Fred Meyer Electronic Centers:
Portland, Albany, Beaverton,
Coos Bay, Salem, Gresham, Medford,
Roseburg, Eugene

Pennsylvania
Data Softique Computer, Pittsburgh

South Carolina
S.F. Enterprises, Sandy Springs
Bytes & Bits, Columbia
Con Current Technology, Travelers Rest
Software Haus, Charleston Heights
Micro Computer Depot, Sumpter

Tennessee
Electronic Warehouse, Decherd
Data World, Knoxville
Computers International, Hixson
Computer King, Knoxville

Texas
Permian Micromart, Odessa
Turnkey Computer Systems, Houston

Utah
Fred Meyer Electronic Centers:
Drem, Ogden, Logan, Salt Lake City
Digital World, West Valley City
Burgoyne Computer Inc.,
South Salt Lake City
Research Micro Systems, Provo

Virginia
Shelken Associates, Inc.,
Fredericksburg

Washington
Fred Meyer Photo Electronic Centers:
Seattle, Tacoma, Bellevue,
Bellingham, Longview

Wisconsin
Micro Age, Milwaukee
Inacomp, Madison

Wyoming
Cheyenne Computer, Cheyenne

THOMSON
A SIGHT FOR SORE EYES.™

Corporate Address
5731 West Slauson Avenue Suite 111 Culver City, CA 90230
(213) 568 1002
(800) 326 0464 (Outside CA)

Children's Software from Springboard.™

The best way to introduce your child to the computer.

Early Games
for Young Children.™

This popular program offers a delightful collection of games designed to teach children ages 2½ to 6 a variety of important basic skills in a fun and friendly way

Children learn to distinguish shapes, recognize letters, say the alphabet, count, add, subtract, even spell their names!

Large, colorful letters, gentle hints and a picture menu help children learn without adult supervision

It's the perfect way to introduce your child to the computer.

Early Games *Easy as ABC*

Satisfaction Guaranteed

At Springboard, we stand behind our products

That's why we offer a full 30-day money-back guarantee Look for these and other Springboard products for all ages at your favorite software store

Now Available For:
Macintosh
Apple II+, IIe, IIc, IBM PC, PCjr, Commodore 64/128

Easy as ABC.™

Learning the alphabet is a wonderful new adventure with Easy as ABC. Five fun games introduce children ages 3 to 6 to letter recognition, alphabetical sequence and upper/lower case letters.

Delightful animals present the alphabet in a variety of ways. Jumping frogs, buzzing bees and soaring rockets fascinate your children as they learn. And multiple skill levels keep them interested so they learn more.

No other program teaches your children the alphabet as thoroughly

SPRINGBOARD™

Springboard Software, Inc • 7808 Creekridge Circle • Minneapolis, MN 55435 • (612) 944-3915

COMPUTE!

JUNE 1986
VOLUME 8
NUMBER 6
ISSUE 73

FEATURES

REVIEWS

COLUMNS AND DEPARTMENTS

THE JOURNAL

NOTE: See page 117 before typing in programs.

TOLL FREE Subscription Order Line
800-247-5470 (In IA 800-532-1272)

GUIDE TO ARTICLES AND PROGRAMS
•
•
128/64/AT/AP/PC/PCjr/AM

AP Apple **Mac** Macintosh, **AT** Atari, **ST,** Atari ST, **V** VIC-20, **64** Commodore 64, **+4** Commodore Plus/4, **16** Commodore 16, **128** Commodore 128, **P** PET/CBM **TI** Texas Instruments, **PC** IBM PC, **PCjr** IBM PCjr, **AM** Amiga •General interest

COMPUTE! Publications, Inc.
Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies
ABC Publishing, President, Robert G. Burton
1330 Avenue of the Americas, New York, New York 10019

COMPUTE! The Journal for Progressive Computing (USPS 537250) is published monthly by COMPUTE! Publications, Inc., 825 7th Ave., New York, NY 10019 USA. Phone (212) 265-8360. Editorial Offices are located at 324 West Wendover Avenue, Greensboro, NC 27408. Domestic Subscriptions: 12 issues, $24. POSTMASTER: Send address changes to **COMPUTE!** Magazine, P.O. Box 10955, Des Moines, IA 50950. Second class postage paid at Greensboro, NC 27403 and additional mailing offices.

ISSN 0194-357X.

Editor's Notes

A New Challenge For ST Programmers

If you're an Atari ST enthusiast, we've got some fantastic news for you.

There's only one catch. We can't tell you what the news is—yet. All we can say is that COMPUTE! is preparing a major surprise that we think you'll enjoy. And to make this surprise as fantastic as possible, we need your help.

If you have access to an ST, and if you're a skillful programmer or writer, we want to see your work. We're looking for ST-oriented articles on a wide variety of topics: tutorials, application programs, utilities, games, educational programs, or almost anything else that we think will be of interest to the several hundred thousand people who already own and use Atari ST computers.

And to break things really wide open, in this instance we're modifying a longstanding COMPUTE! policy regarding program submissions. Up to now, for the most part, we've restricted the programs we publish to either BASIC or machine language, and we've also restricted their length. This has forced us to turn down some otherwise outstanding submissions, but we've had good reasons for it.

These reasons have to do with the realities of magazine publishing. We've insisted that programs be written in BASIC or ML because those are the only two languages that everybody owns. Practically every personal computer comes with BASIC, and ML is every computer's native language. If we publish a program written in some other language—such as Pascal, C, COMAL, Forth, or whatever—the number of readers who can use the program suddenly shrinks to a tiny minority. Realistically, a magazine that wants to stay in business has to appeal to a majority of its readers most of the time. Thus, we've avoided programs written in "nonstandard" languages, although it's been frustrating to all of us.

A related problem is the restriction we've traditionally placed on the length of programs. Again, this has to do with an unpleasant side of magazine publishing. Sadly, we've had to reject some excellent programs merely because they were too long to print. There's a limit to how much typing a reader is willing to undertake, even to get an exceptional program. Recently we've stretched this limit near the bursting point. We believe that programs like our *SpeedScript* word processor and *SpeedCalc* spreadsheet—with versions for Commodore, Atari, and Apple computers—are the best applications ever offered by a computer magazine. But both programs were written entirely in ML and required readers to spend many hours typing in thousands of numbers. Our MLX machine language entry utility is a partial solution. So is our COMPUTE! DISK. But we can't assume every reader is going to buy the disk, so we still have to restrict the length of programs to keep them accessible to all of our readers.

The new generation of high-powered, low-cost personal computers—exemplified by the Atari ST series—is allowing us to rethink our approach to program publishing. As the hardware grows more powerful, so does the software. The programs printed in magazines have to keep up, too. Some people go so far as to say that the days of program-oriented magazines are coming to an end. We strongly disagree. Consistently, reader feedback tells us that our programs and programming tutorials are the most popular features of our magazines. We feel that many useful programs can still be written in BASIC, and that BASIC will continue to be the language of choice for home programmers for some time to come. But to turn out really exceptional pieces of work, more and more programmers will be forced to turn to alternatives—particularly compilers. And their programs will grow larger and larger.

To meet this challenge, we're taking an exciting new approach. The details of this approach are part of the surprise we're preparing. For now, however, we can say this much:

We'll consider Atari ST program submissions written in practically any programming language you want. Have you written a utility in C for designing character fonts? Have you discovered a way to implement drop-down menus in ST BASIC? Have you written a general-purpose database manager in Prolog? Or an educational program in Pascal? Or a terminal program in Forth? Or an arcade-style game in machine language? Or a text editor in Modula-2?

Whatever it is, we'd like to see it. But don't get the idea that we're not picky. As always, we're interested in obtaining only the best-quality programs and articles we can find. If necessary, these programs can be much longer than ones we'd ordinarily publish in printed form. Of course, we still prefer to see programs which are as efficiently written as possible, so don't get carried away.

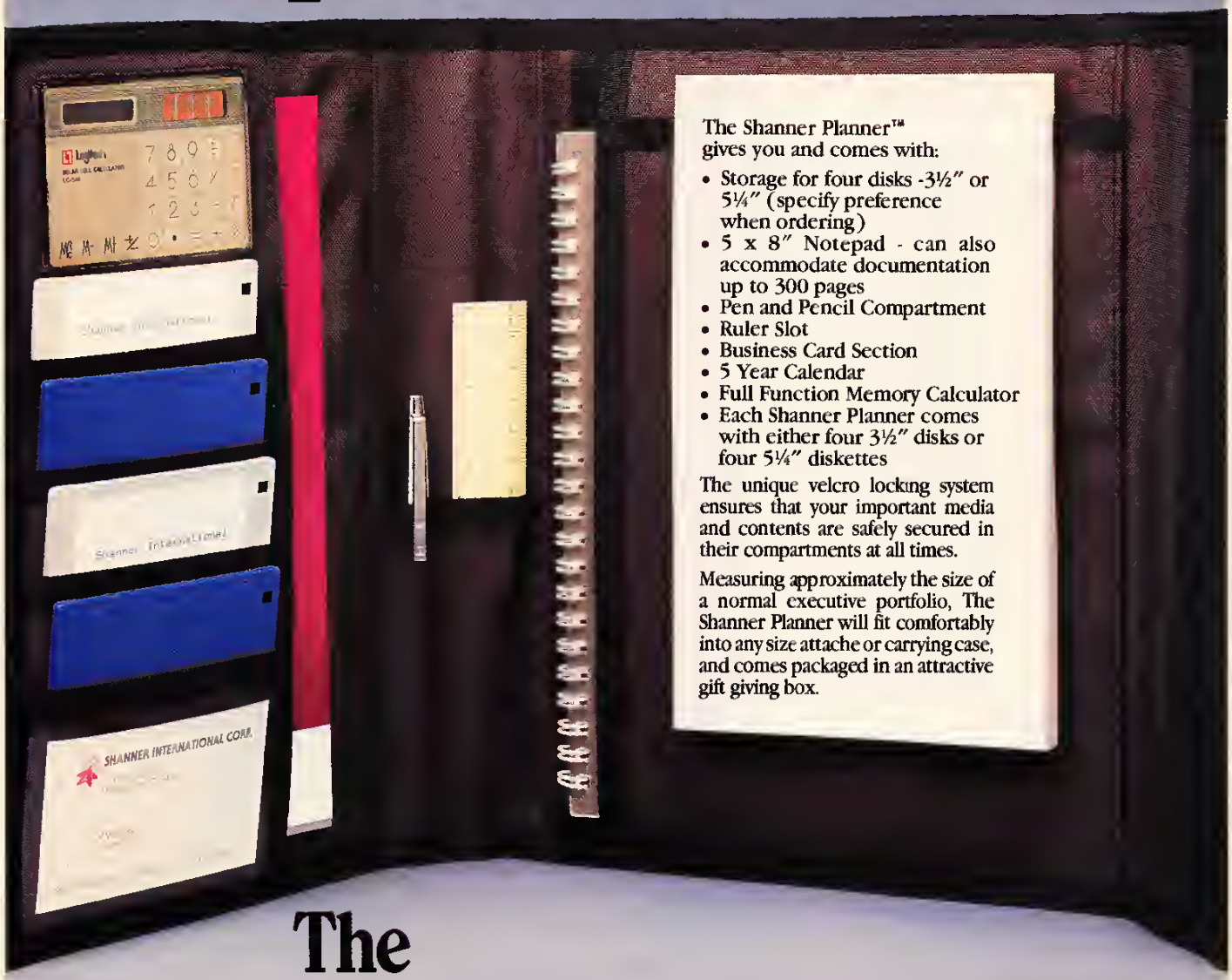
There's only one restriction: *The executable object code of the program must be legally usable by someone who doesn't own a copy of the language.* For instance, if you write a program with a compiled language, the compiled code must be a self-standing run-time package that anyone can load and run, whether or not they own the compiler. And we must be able to legally distribute the run-time package without becoming entangled in licensing fees and so forth. If you aren't sure about this, check with the company which produces the language.

Aside from this minor restriction, the gates are wide open. As a further incentive, we can hint that because of the way we'll be publishing these programs, some significant royalties may be in store for those whose work is accepted.

This is going to be an exciting experiment for all ST enthusiasts—readers, programmers, and those of us at COMPUTE!. Let's all make it a success.

Tom R. Halfhill, Editor

It speaks for itself.



The Shanner Planner™ gives you and comes with:

- Storage for four disks -3½" or 5¼" (specify preference when ordering)
- 5 x 8" Notepad - can also accommodate documentation up to 300 pages
- Pen and Pencil Compartment
- Ruler Slot
- Business Card Section
- 5 Year Calendar
- Full Function Memory Calculator
- Each Shanner Planner comes with either four 3½" disks or four 5¼" diskettes

The unique velcro locking system ensures that your important media and contents are safely secured in their compartments at all times.

Measuring approximately the size of a normal executive portfolio, The Shanner Planner will fit comfortably into any size attache or carrying case, and comes packaged in an attractive gift giving box.

The Shanner Planner™

$39.95

Dealer Inquiries Accepted
Call 800/828-6637

Remember the expression "A picture is worth a thousand words?" Take a second look. The Shanner Planner™ is the first ever full system portfolio created specifically for the computer user. Designed to address the individual needs of the computer industry, The Shanner Planner is tastefully constructed of long lasting, durable, textured nylon *(it's the in thing today)* to maintain its smart appearance over time.

SHANNER INTERNATIONAL CORP.

The Shanner Planner™ is a creation of, marketed worldwide by, and is a Trademark of SHANNER INTERNATIONAL CORP

Please send me _____ Shanner Planner(s)

I am enclosing $ _____
(CA residents add $2 60 tax per Planner)
Please make check or money order payable to
Shanner International Corp
453 Ravendale Drive
Mountain View, CA 94030

Canadian Orders
Canadian cost per Planner is $49 95 (Ontario residents add $3 50 PST per Planner) Please direct orders to
Shanner Sales & Marketing, Inc
450 Garyray Drive
Weston, Ontario

Name _____

Address _____

City _____ State _____ Zip _____

For all orders Please add $3 00 per Planner for shipping and handling C O D 's and purchase orders will not be accepted Please allow 21 days for delivery

Publisher	James A. Casella
Founder/Editor In Chief	Robert C. Lock
Senior Editor	Richard Mansfield
Managing Editor	Kathleen Martinek
Executive Editor	Selby Bateman
Editor	Tom R. Halfhill
Assistant Editor	Philip Nelson
Production Director	Tony Roberts
Production Editor	Gail Cowper
Editor, COMPUTEI's GAZETTE	Lance Elko
Technical Editor	Ottis R. Cowper
Assistant Technical Editor	George Miller
Program Editor	Charles Brannon
Assistant Editor, COMPUTEI's GAZETTE	Todd Heimarck
Assistant Features Editor	Kathy Yakal
Programming Supervisor	Patrick Parrish
Editorial Programmers	Tim Victor, Kevin Mykytyn
Research/Copy Editor	Joan Rouleau
Submissions Reviewer	Mark Tuttle
Programming Assistants	David Florance, David Hensley
Executive Assistant	Debi Nash
Administrative Assistants	Julia Fleming, Iris Brooks, Mary Hunt, Sybil Agee
Associate Editors	Jim Butterfield, Toronto, Canada Harvey Herman, Greensboro, NC Fred D'Ignazio, Birmingham, AL David Thornburg, Los Altos, CA
Contributing Editor	Bill Wilkinson
COMPUTEI's Book Division Editor	Stephen Levy
Assistant Editors	Gregg Keizer, Ann Davies
Director, Book Sales & Marketing	Steve Voyatzis
Production Manager	Irma Swain
Art & Design Director	Janice R. Fary
Assistant Editor, Art & Design	Lee Noel
Mechanical Art Supervisor	De Potter
Artists	Debbie Bray, Dabney Ketraw
Typesetting	Terry Cash, Carole Dunton
Illustrator	Harry Blair
Director of Advertising Sales	Peter Johnsmeyer
Associate Advertising Director	Bernard J. Theobald, Jr.
Production Coordinator	Kathleen Hanlon
Promotion Assistant	Caroline Dark
Customer Service Manager	Dane Longo
Dealer Sales Supervisor	Orchid Tamayo
Individual Order Supervisor	Cassandra Green
Receptionist	Anita Armfield
Warehouse Manager	John Williams
Data Processing Manager	Leon Stokes

James A. Casella, President
Richard J. Marino, Vice President, Advertising Sales

COMPUTE! Publications, Inc. publishes

COMPUTE!
The Journal For Progressive Computing

COMPUTE!'s GAZETTE
For VIC-20 And Commodore 64 Personal Computers

COMPUTEI Books

COMPUTE!'s GAZETTE DISK

COMPUTEI's Apple Applications Special

Editorial offices	324 West Wendover Avenue Suite 200 Greensboro, NC 27408 USA
Corporate offices	825 7th Avenue New York, NY 10019 212-265-8360
Customer Service	800-346-6767 (In NY 212-887-8525)
Hours	9:30 A.M.–4:30 P.M. Monday–Friday

Coming In Future Issues

Hex War: Strategic Simulation For Commodore 64, 128, Atari 400/800/XL/XE, Atari ST And Others

The Top Five Public-Domain Programs For Your Computer

Super-Wide Printouts For Commodore: A Better Way To Print Spreadsheets And Wide Documents

Atari Sound Development System For Atari 400/800/XL/XE

The Screen Machine II Menu-Driven Drawing Program For IBM PC, PCjr, Compatibles

Apple Printer Master Get Full Power From Your Printer

Subscription Orders

COMPUTEI
P.O. Box 10954
Des Moines, IA 50340

TOLL FREE Subscription Order Line
800-247-5470
In IA 800-532-1272

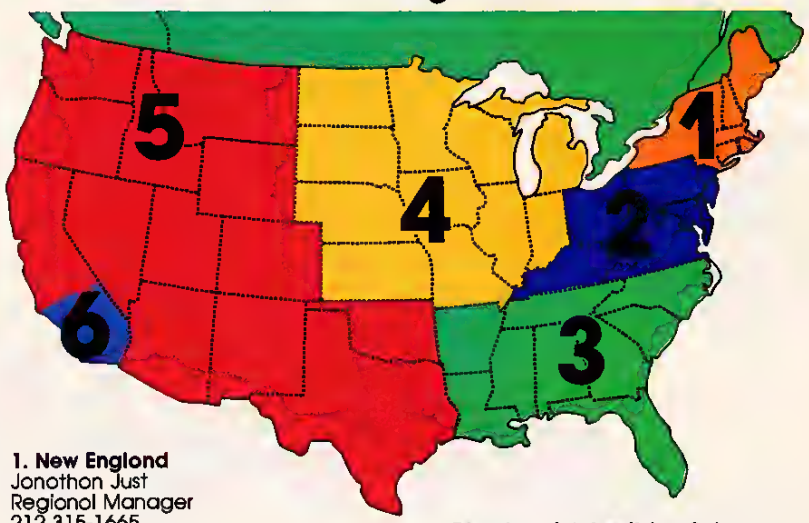
COMPUTEI Subscription Rates (12 Issue Year):

US	(one yr.)	$24
	(two yrs.)	$45
	(three yrs.)	$65
Canada and Foreign Surface Mail		$30
Foreign Air Delivery		$65

Member ABC — Audit Bureau of Circulations

MPA — Magazine Publishers Association

Advertising Sales



1. New England
Jonathon Just
Regional Manager
212-315-1665

2. Mid Atlantic
Jonathon Just
Regional Manager
212-315-1665

3. Southeast & Foreign
Harry Blair
919-275-9809

4. Midwest
Gordon Benson
312-362-1821

5. Northwest/ Mountain/Texas
Phoebe Thompson
Dani Nunes
408-354-5553

6. Southwest
Ed Winchell
213-378-8361

Director of Advertising Sales:
Peter Johnsmeyer

Associate Advertising Director:
Bernard J. Theobald, Jr.

COMPUTEI Home Office 212-887-8460.

Address all advertising materials to:
Kathleen Hanlon
Advertising Production Coordinator
COMPUTEI Magazine
324 West Wendover Avenue
Suite 200
Greensboro, NC 27408

GET FREE SOFTWARE WITH BOWATER PAPER AND BROWN DISC SCHOLAR DISKETTES!

"Ean, our computer scholar, has helped my wife, Brandi, understand how many valuable incentives come with Bowater computer paper and Brown Disc *SCHOLAR* diskettes. Ean already has Ben printing banners and graphics from his new software programs given free with these products.

"We get free Brown diskettes with the purchase of Bowater paper and free Bowater paper with the purchase of Brown Disc's *SCHOLAR* diskettes. And now, WE CAN CHOOSE FROM 3 DIFFERENT SOFTWARE PROGRAMS TO GET ONE FREE WITH EACH 1000 SHEET PACKAGE OF BOWATER COMPUTER PAPER OR WITH A TEN PACK OF BROWN DISC *SCHOLAR* DISKETTES.

"Even Ryan enjoys coloring the graphics that Ben printed out, and Brandi prefers the children use the computer to entertain themselves. . . rather than watch TV.

"Best of all, the kids are working with the new software to surprise me. Thanks Bowater and Brown Disc. . . you really give a family REAL VALUE."

Phil Wismer
Husband, father and business professional

You too can receive valuable incentives by purchasing Bowater computer paper and Brown Disc *SCHOLAR* diskettes. **See the listing on the reverse side of this ad to find our dealers in your area.**

BOWATER
BOWATER COMPUTER FORMS INC.

P.O. Box 869020
Plano, TX 75086-9020
1-800/527-3412
In Texas Call Collect 214/578-2000

SCHOLAR
by brown disc

1110 Chapel Hills Drive
Colorado Springs, CO 80918
1-800/654-4871
303/593-1015 Inside CO

CLIP THIS PAGE TO FIND YOUR CLOSEST BOWATER AND BROWN DISC DEALERS CARRYING OUR NEW FREE SOFTWARE OFFER!

BOWATER COMPUTER FORMS INC.

CALIFORNIA
- TECHNICA
  ALBANY — 415/525-9404
- ALKO OFFICE SUPPLY
  BERKELEY — 415/848-3356
- SIERRA COMPUTER SUPPLIES
  CAMPBELL — 408/374-7810
- MICRO AGE COMPUTER STORE
  CONCORD — 415/680-1489
- WORLD OF COMPUTERS
  CONCORD — 415/825-1571
- WORLD OF COMPUTERS
  DUBLIN — 415/829-7211
- KINGS OFFICE PRODUCTS
  FAIRFIELD — 707/427-2933
- DEAN MARK OFFICE SUPPLY INC
  FREMONT — 415/791-0344
- DEAN MARK OFFICE SUPPLY INC
  FREMONT — 415/490-7448
- LEDEENS COMPUTER & OFFICE SUPPLY
  FREMONT — 415/793-3600
- DISCOUNT OFFICE SUPPLY
  MT. VIEW — 415/962-8900
- COMPUTER STORE OF OAKLAND
  OAKLAND — 415/763-7900
- WORLD OF COMPUTERS
  OAKLAND — 415/451-0212
- UNIVERSITY STATIONERS
  PALO ALTO — 415/326-7970
- DEAN MARK OFFICE SUPPLY INC
  PLEASANTON — 415/426-0282
- ALAMO COMPUTER
  SAN JOSE — 408/227-0691
- COMPUTER SUPPLIES UNLIMITED
  SAN JOSE — 408/265-0200
- DEAN MARK OFFICE SUPPLY INC
  SAN JOSE — 408/436-1833
- TELEMEDIA
  SAN JOSE — 408/435-7330
- M & M OFFICE SUPPLY
  SAN RAMON — 415/829-7283
- SANTA ROSA COMPUTER
  SANTA ROSA — 707/528-1291
- VORHEE'S COMPUTERS
  SANTA ROSA — 707/575-1419
- PALACE ART & OFFICE SUPPLY
  SOQUEL — 408/476-3799
- FRYS ELECTRONICS
  SUNNYVALE — 408/733-1770
- SUNNYVALE STATIONERS
  SUNNYVALE — 408/739-7000
- WORLD OF COMPUTERS
  WALNUT CREEK — 415/946-1070

COLORADO
- MAGNA TRONICS WEST
  GOLDEN — 303/237-5550

FLORIDA
- ACS SERVICES INC
  TALLAHASSEE — 904/681-6566

GEORGIA
- B & W ENTERPRISES
  WAYCROSS — 912/285-7206

INDIANA
- DIVERSIFIED DATA SYSTEMS
  INDIANAPOLIS — 317/297-5177

KANSAS
- WESTERN OFFICE SUPPLY
  COLBY — 913/462-3923
- COMPUTER GIN
  DODGE CITY — 316/225-2628
- THOROUGHBRED COMPUTERS
  EMPORIA — 316/342-0221
- BAAN TECH BUSINESS PRODUCTS INC
  GARDEN CITY — 316/275-7419
- COMPUTER TEK OFFICE PRODUCTS
  GARDEN CITY — 316/275-7514
- COMPUTERLAND OF GREAT BEND
  GREAT BEND — 316/792-3524
- WESTERN OFFICE SUPPLY
  HAYES — 913/625-0027
- COMPUTERLAND OF HUTCHINSON
  HUTCHINSON — 316/662-6032
- THE COMPUTER GENERATION
  INDEPENDENCE — 316/331-0903
- COMPUTERARK
  LAWRENCE — 913/841-0094
- FORESIGHT SOLUTIONS dba COMPUTER OUTLET
  LAWRENCE — 913/842-7526
- CUSTOM BUSINESS SYSTEMS INC
  MANHATTEN — 913/537-4306
- ADVANCE COMPUTER TECH
  McPHERSON — 316/241-5093
- THE COMPUTER ROOM
  McPHERSON — 316/241-8800
- COMPUTERLAND OF SALINA
  SALINA — 913/823-1555
- MICROAGE OF SALINA
  SALINA — 913/823-7596
- COMPUTRAIN INC
  TOPEKA — 913/272-6800
- MEEKS INC
  TOPEKA — 913/272-4750
- STAUFFER EQUIPMENT & SUPPLY CO
  TOPEKA — 913/295-1157
- THOROUGHBRED COMPUTERS
  TOPEKA — 913/232-8089
- SOUTH CENTRAL BUSINESS FORMS INC
  WICHITA — 316/264-0220
- WILBUR E. WALKER CO. INC
  WICHITA — 316/267-2231
- ABEL SERVICES
  WINFIELD — 316/221-7577
- VARIETY HAUS
  ULYSSES — 316/356-3820

KENTUCKY
- NATIONAL OFFICE SUPPLY
  COVINGTON — 606/431-5100
- SUPERIOR PAPER COMPANY
  LOUISVILLE — 502/583-1647

MASSACHUSETTS
- COMPUTER DISCOUNT
  NORFOLK — 617/528-4186

MICHIGAN
- RETAIL COMPUTER CENTER, INC
  BIRMINGHAM — 313/644-4820
- TIERRA
  CLARKSTON — 313/625-2511
- RETAIL COMPUTER CENTER, INC
  FARMINGTON HILLS — 313/626-3240
- RETAIL COMPUTER CENTER, INC
  GARDEN CITY — 313/422-2570
- OFFTECH
  GRAND RAPIDS — 616/451-8561
- MICRO WORLD COMPUTER CENTER
  LIVONIA — 313/427-0102
- MICRO WORLD COMPUTER CENTER
  MILFORD — 313/685-7766
- COMMUNIGRAPHIX, INC
  ROSEVILLE — 313/445-8970
- MICRO WORLD COMPUTER CENTER
  ROSEVILLE — 313/771-3600
- MICRO WORLD COMPUTER CENTER
  SOUTHGATE — 313/285-0101
- LAKES INC
  UNION LAKE — 313/360-0681

MINNESOTA
- BUDGET SOFTWARE
  CRYSTAL — 612/535-4369
- COMPUTER SATISFACTION
  ELK RIVER — 612/441-4225
- TEAM ELECTRONICS
  MARRATO — 507/387-7937
- PARAGON COMPUTERS
  ROCHESTER — 507/205-1999
- SOFTWARE CENTRE INTERNATIONAL
  ROSEVILLE — 612/631-3500

NEW JERSEY
- MOI COMPUTER STORE
  ENGLISHTOWN — 201/462-4600
- PERIPHERALS PLUS INC
  HOWELL — 201/363-6270
- STONEHENGE COMPUTER
  OLMMIT — 201/277-1020
- COMPUTER SERVICES CO
  UPPER SADDLE RV — 201/327-8811
- COMPUTER OUTLET
  WESTWOOD — 201/666-8808

NEW MEXICO
- ANOATA
  LOS ALAMOS — 505/662-0031
  800/552-0070 in the mountain time zone

NEW YORK
- COMPUTER SOFT CENTER
  BREWSTER — 914/279-5001
- CORBIT BUSINESS MICRO SYSTEMS
  BRIGHTWATERS — 516/665-7811
- PRIME TIME COMPUTER
  BROOKLYN — 718/232-7770
- SYSUT COMPUTERS
  BNOORLYN — 718/743-8303
- VIDEO COMPUTER BIN
  BNOORLYN — 718/241-1993
- GA COMPUTER PRODUCTS
  BUFFALO — 716/854-0004
- MONROE CAMERA & COMPUTER
  MONROE — 914/782-8525
- COMPUTER OUTLET
  NEW CITY — 914/639-3000
- CIRO PRINTING & OFFICE SUPPLIES
  NEW YONK — 212/406-1323
- GERSTEN BUSINESS MACHINE
  NEW YORK — 212/764-6500
- HI TEC COMPUTERS
  NEW YORK — 212/374-1008
- INPUT COMPUTER
  NEW YORK — 212/725-9561
- SIMPLY COMPUTER & OFFICE PRODUCTS
  NEW YORK — 212/732-9336
- STATEWIDE OFFICE SUPPLY
  NEW YORK — 212/213-0770
- VENUS OFFICE SUPPLY CORP
  NEW YORK — 212/477-2000
- COMPUTER CENTER
  ROCHESTER — 716/262-3166
- GA COMPUTER PRODUCTS
  ROCHESTER — 716/342-0602
- GA COMPUTER PRODUCTS
  SYRACUSE — 315/425-1414
- COMPURITE CENTRE
  WHITE PLAINS — 914/946-0900

NORTH CAROLINA
- MICRO BASICS
  CHARLOTTE — 704/527-1370

OHIO
- TETRA SYSTEMS
  AKRON — 216/666-2978
- MID-WEST MICRO
  ST. PANIS — 513/663-5408
- R.L. STEINMAN & SON
  ARLINGTON — 419/365-5555
- SOFTWAIRE CENTRE
  CANTON — 216/492-9163
- COPY SOURCE
  CINCINNATI — 513/489-6821
- IMAGING SUPPLIES
  CINCINNATI — 513/871-2525
- MIDWEST COMPUTER SUPPLIES
  CINCINNATI — 513/481-0020
- SCHEAR COMPUTER SUPPLIES
  CINCINNATI — 513/729-5522
- DISKOUNT SOFTWARE
  COLUMBUS — 614/231-9910
- BEST LITTLE WAREHOUSE
  CUYAHOGA FALLS — 216/920-1136
- CAD CAM
  DAYTON — 513/293-3301
- EXPRESS BUSINESS FORMS
  DAYTON — 513/866-1788
- PC NETWORK SYSTEMS
  DAYTON — 513/435-2256
- TETRA SYSTEMS
  DAYTON — 513/299-3476
- COMPUTER WAREHOUSE
  EUCLID — 216/481-5515
- SOFTWARE CONNECTION
  MAYFIELD HEIGHTS — 216/473-2722
- VALCOM
  MAYFIELD HEIGHTS — 216/449-4030
- FREDERICK COMPUTER PRODUCTS
  MIDDLETOWN — 513/424-3233
- COMPUTER SHOWCASE
  NILES — 216/652-2571
- SOFTWAIRE CENTER
  NORTH OLMSTED — 216/734-4755
- BIG BYTE COMPUTERS
  POLAND — 216/758-0009
- COMPUTER DISCOUNTS
  REYNOLDSBURG — 614/863-6701

PENNSYLVANIA
- WEAN ASSOCIATES
  CONSHOHOCKEN — 215/834-0866
- SUBURBAN OFFICE SUPPLIERS
  LANSDALE — 215/362-6510

SOUTH CAROLINA
- S T D P INC
  CAYCE — 803/796-5600
- MICRO COMPUTER DEPOT
  SUMTER — 803/775-5166

TENNESSEE
- ECLECTIC SOLUTIONS CORPORATION
  CLEVELAND — 615/336-3658

TEXAS
- SHRIVER OFFICE SUPPLY
  ALICE — 512/664-2630
- DICK OFFICE SUPPLY
  AUSTIN — 512/258-7960
- INFORMATION PROCESSING SUPPLY
  AUSTIN — 512/445-7223
- KELLY OFFICE PRODUCTS
  CEDAR PARK — 512/258-1656
- A & D OFFICE SUPPLY
  CORPUS CHRISTI — 512/883-1865
- EL PASO MICRO MART INC
  EL PASO — 915/595-1100
- STEELMAN OFFICE PRODUCTS
  GONZALES — 512/672-9535
- DICK OFFICE SUPPLY
  HARLINGEN — 512/428-1221
- COMPUTERS, INC
  HOUSTON — 713/556-1331
- EXCEL BUSINESS SYSTEMS, INC
  HOUSTON — 713/952-0236
- COOK OFFICE MACHINES
  McALLEN — 512/686-9576
- DICK OFFICE SUPPLY
  McALLEN — 512/682-6306
- COMAL OFFICE PRODUCTS
  NEW BRAUNFELS — 512/625-9108
- PERMIAN MICRO MART
  ODESSA — 915/367-6179
- ALL AMERICAN OFFICE SUPPLY
  PLANO — 214/424-0524
- WILDCAT COMPUTING, INC
  PLANO — 214/424-3503
- HOUSE OF SUPPLIES
  RAYMONDVILLE — 512/689-2524
- AB COMPUTER ELECTRONICS CENTER
  ROUND ROCK — 512/244-2090
- THE COMPUTER STORE
  SAN ANGELO — 915/942-7505
- AMERICAN OFFICE PRODUCTS
  SAN ANTONIO — 512/654-8808
- AZTEC BUSINESS FORMS
  SAN ANTONIO — 512/654-7969
- BUSINESS WORLD
  SAN ANTONIO — 512/820-0103
- PAUL ANDERSON CO
  SAN ANTONIO — 512/734-0111
- PREMIER BUSINESS PRODUCTS
  SAN ANTONIO — 512/600-4149
- PROFESSIONAL BUSINESS PRODUCTS
  SAN ANTONIO — 512/822-5544
- TA OFFICE SUPPLY
  SAN ANTONIO — 512/377-3500
- WAGNER BROTHERS COMPUTER STORE
  SAN ANTONIO — 512/657-2012
- COMMERCIAL OFFICE PRODUCTS
  SAN MARCOS — 512/353-7477
- KLIPPLE BUSINESS PRODUCTS
  SAN MARCOS — 512/396-5055
- STEELMAN OFFICE PRODUCTS
  SEGUIN — 512/372-3950
- VICTORIA OFFICE EQUIPMENT
  VICTORIA — 512/573-4371

BROWN DISC MANUFACTURING, INC.

TO LOCATE SCHOLAR DEALERS IN YOUR AREA CALL:

NATIONWIDE
- ALVIN/MODERN SCHOOL SUPPLY
  1-800-234-2329
  CT 203/243-8991
- RICHARD YOUNG OFFICE PRODUCTS
  1-800-325-0136
  FL 305/979-3100
- NASCORP INC
  1-800-321-3883
  OH 216/774-1831

NORTHWEST
- ECZEL CORP
  1-000-431-2489
  WA 206/872-7636

ROCKY MOUNTAIN
- COMPUTER SERVICES BROKER
  1-950-1088 at dial tone 770-185
  CO 303/697-0561

ALABAMA
- DUCHESS COMPUTER PRODUCTS
  GRAND BAY — 205/865-6324
- INTERSTATE SCHOOL SUPPLY & EQUIPMENT COMPANY OF ARKANSAS TENNESSEE ALABAMA
  1-800-222-8600
  MS 601/948-8600

ARKANSAS
- INTERSTATE SCHOOL SUPPLY & EQUIPMENT COMPANY OF ARKANSAS TENNESSEE ALABAMA
  1-800-222-8600
  MS 601/940-0600

COLORADO
- COLBORN'S
  DENVER — 303/770-1220

FLORIDA
- GRANDS MART
  N. MIAMI — 305/624-5400
- COLLEGE MANAGEMENT SERVICE INC
  FL 305/703-3100
- DUCHESS COMPUTER PRODUCTS
  AL 205/865-6324
- INTERSTATE SCHOOL SUPPLIERS
  1-800-241-4336
  GA 404/997-6714

GEORGIA
- DUCHESS COMPUTER PRODUCTS
  AL 205/865-6324
- INTERSTATE SCHOOL SUPPLIERS
  1-800-241-4336
  ATLANTA — 404/997-6714

ILLINOIS
- DUCHESS COMPUTER PRODUCTS
  AL 205/865-6324
- MIDWEST COMPUTER SUPPLY
  WEST CHICAGO — 312/231-1112

INDIANA
- COLLEGE MANAGEMENT SERVICE INC
  FL 305/703-3100
- MIDWEST COMPUTER SUPPLY
  IL 312/231-1112

KENTUCKY
- MENTOR SYSTEMS
  LEXINGTON — 606/231-0519

LOUISIANA
- INTERSTATE COMPANIES OF LOUISIANA
  1-800-272-9800
  BATON ROUGE — 504/387-5131
  NEW ORLEANS — 504/561-9500

MARYLAND
- DUCHESS COMPUTER PRODUCTS
  AL 205/865-6324

MISSISSIPPI
- DUCHESS COMPUTER PRODUCTS
  AL 205/865-6324
- MISSISSIPPI SCHOOL SUPPLY COMPANY OF MISSISSIPPI
  1-800-222-8600
  JACKSON — 601/948-8600

MONTANA
- COLBORN'S
  BILLINGS — 406/245-3158

NEBRASKA
- COLLEGE MANAGEMENT SERVICE INC
  FL 305/703-3100

NEW JERSEY
- COLLEGE MANAGEMENT SERVICE INC
  FL 305/703-3100

NEW MEXICO
- COLBORN'S
  ALBUQUERQUE — 505/292-3225
- SPARTAN II ELECTRONICS INC
  ALBUQUERQUE — 505/345-3479

NORTH DAKOTA
- COLBORN'S
  GRAND FORKS — 701/746-5441

SOUTH CAROLINA
- INTERSTATE SCHOOL SUPPLIERS
  1-800-241-4336
  GA 404/997-6714

TENNESSEE
- DUCHESS COMPUTER PRODUCTS
  AL 205/865-6324
- INTERSTATE SCHOOL SUPPLY & EQUIPMENT COMPANY OF ARKANSAS TENNESSEE ALABAMA
  1-800-222-0600
  MS 601/948-0600

TEXAS
- SPARTAN II ELECTRONICS INC
  NM 505/345-3479

WYOMING
- COLBORN'S
  CASPER — 307/265-5026

Ask about other BROWN DISC AND BOWATER SPECIAL OFFERS. BROWN DISC ALSO GIVES YOU 100 SHEET PACKAGES OF FREE BOWATER COMPUTER PAPER WITH THE PURCHASE OF A SCHOLAR 2 PACK.

AND,

BOWATER GIVES YOU 1 FREE BROWN DISKETTE WITH 500 SHEETS AND 2 FREE BROWN DISKETTES WITH 2500 SHEETS OF BOWATER COMPUTER PAPER.

So, see our dealers today to receive these value packed products!

Readers Feedback

The Editors and Readers of COMPUTE!

If you have any questions, comments, or suggestions you would like to see addressed in this column, write to "Readers' Feedback," COMPUTE!, P.O. Box 5406, Greensboro, NC 27403. Due to the volume of mail we receive, we regret that we cannot provide personal answers to technical questions.

SpeedScript's Lineage

What are the differences between *SpeedScript* 3.0, 3.1, 3.2, and so on?

Leo Mitchener

Here's the genealogy of SpeedScript *for the Commodore 64: The original 64* SpeedScript *(now called version 1.0) appeared in the January 1984 issue of* COMPUTE!'s GAZETTE. *A slightly modified version (1.1) appeared in* COMPUTE!'s Second Book of Commodore 64. *The next major update,* SpeedScript *2.0, appeared only on the premier* GAZETTE DISK *in May 1984. Like the original, its title screen did not include a version number; however, it can be distinguished from other versions by its custom character set and help screen.*

Version 3.0 made its debut in the March 1985 issue of COMPUTE! *and on the special* COMPUTE! DISK *for that month. It can easily be distinguished from its predecessors because the command line says* SpeedScript 3.0. *Corrections for several minor bugs were published in the May 1985 "CAPUTE!" column. With these enhancements, the title on the screen indicates version 3.1. It was this version which appeared in the book* SpeedScript: The Word Processor for the Commodore 64 and VIC-20, *and on the companion disk for that book. Further corrections—most notably a fix for an underlining bug—appeared in the article "*SpeedScript *3.0 Revisited" in the December 1985 issue of* COMPUTE!; *these enhancements changed the version number on the screen to 3.2. Version 3.2 also appeared on the January 1986* COMPUTE! DISK. *The corrections in the December article included the changes from version 3.1, so it is possible to upgrade directly from 3.0 to 3.2.*

As in many other areas of personal computing, there isn't any official rule that dictates how program versions are to be numbered. For SpeedScript *we've followed what seems to be the most common convention. In general, a whole number difference (such as 2.0 versus 3.0) signals a major enhancement, while a fractional change (3.0 versus 3.2) indicates minor enhancements. Unless otherwise indicated, a reference to one member of a group is also applicable to the others. We usually use* SpeedScript 3.0 *to refer to all members of the version 3 family: 3.0, 3.1, and 3.2. For example, the 3.0 version of the POKEs given in the January "Readers' Feedback" to make* SpeedScript *default to disk or tape also works for 3.2, even though this was not stated explicitly.*

For a description of how SpeedScript *3.0 differs from previous versions in terms of features, see the article in the March 1985 issue of* COMPUTE!.

The VIC-20 version of SpeedScript 3.0 *appeared in the April 1985 issue of* COMPUTE!. *The Atari and Apple versions of* SpeedScript *start with version 3.0 and made their debut in the May 1985 and June 1985 issues of* COMPUTE!, *respectively.*

Machine Language Delays

I have recently written a program in 6502 machine language for the VIC-20. I want to have a one- or two-second pause between the title screen and the main program, but I don't know how to make one.

Stephen Brown

One way to create a delay in machine language (ML) is to use a do-nothing loop much as you would in BASIC. For instance, the BASIC loop shown here pauses for about one second on a VIC:

```
FOR TD=1 TO 1000:NEXT
```

A similar machine language loop looks like this:

```
        LDY #0
WAIT    DEY
        BNE WAIT
        RTS
```

This loop creates a delay, but only for a fraction of a second. To produce a longer delay, you could use two nested loops:

```
        LDY #0
        LDX #0
WAIT    DEY
        BNE WAIT
        DEX
        BNE WAIT
        RTS
```

This loop delays for about a second. For longer delays you can use more nested loops combining different memory locations and registers. Some computers have a built-in clock that's available for the same purpose. On the Commodore 64 and VIC-20, for instance, location 162 is incremented every 1/60 second by the computer's hardware interrupt routine. To create a delay with the built-in clock, store a zero in location 162, then wait until it reaches the number of seconds you want to delay divided by 60. This short routine creates a three-second delay:

```
        LDA #0
        STA 162
WAIT    LDA 162
        CMP #180
        BNE WAIT
        RTS
```

Changing Apple Proofreader's Checksum

I am using an Apple IIe with a color TV as a monitor. One problem with the TV is that reverse characters are difficult if not impossible to read. Is there any way to modify the "Apple Automatic Proofreader" so the checksum numbers appear normal instead of reverse? I am not the best typist in the world and was delighted to find a Proofreader program. But the checksum numbers are so hard to read that I can't use it at all.

Robert A. Love

It's easy to defeat the reverse video effect for Apple computers. Run the Proofreader as usual, then enter this line in direct mode (without a line number):

POKE 804,176: POKE 806,186: POKE 822,176: POKE 824,186

The checksum numbers appear in the usual screen location in normal video. Since this modification makes the checksum harder to distinguish from other numbers on the screen, you probably won't want to make this change unless it's absolutely necessary.

Scrolling Atari Messages

I am an Atari 1200XL owner. I would like to know how to move a message like 1=LOAD 2=LOCK 3=UNLOCK across the screen.

Bobby Chan

The following BASIC program scrolls any message up to 100 characters across the top of the screen. The variable MESS$ in line 20 contains the message to be printed. You can reposition the scrolling message to any line on the screen by changing the POSITION statement in line 30.

```
KK 5 DIM MESS$(100),TEMP$(200):L=1
EC 10 PRINT "{CLEAR}":POKE 82,0:POKE 752,1
JO 20 MESS$="TYPE LETTER TO RUN, OR 1=LOAD 2=LOCK 3=UNLOCK 4=EXIT... ":N=LEN(MESS$)
FE 25 TEMP$(1,N)=MESS$:TEMP$(N+1,2*N)=MESS$
DB 30 POSITION 0,0:PRINT TEMP$(L,L+39)
AM 40 L=L+1:IF L>2*N-40 THEN L=1
EB 50 FOR TD=1 TO 50:NEXT TD:GOTO 30
```

IBM BASIC Directory

Can you tell me how to read and display the disk directory on an IBM PC from within a BASIC program?

Kamal Ashour

There are two simple ways to approach this. The first is simply to print the directory to the screen at the appropriate time in your BASIC program. A second method would be to read the directory into a string array for use by your program at some later point. Here's a short routine that employs the first method:

```
EG 100 REM FSPEC$="A:*.*":GOTO 140
ND 110 PRINT:PRINT"Select drive: (";:COLOR 16,15:PRINT"A B";:COLOR 7,0:PRINT CHR$(29)CHR$(29)"/"CHR$(28)")"
HB 120 DRIVE$=INKEY$+":":A=ASC(DRIVE$):IF (A OR 32)<97 OR (A OR 32)>98 THEN 120
IF 130 DRIVE$=CHR$(A AND 223)+":":FSPEC$=DRIVE$+"*.*"
GG 140 ON ERROR GOTO 150:FILES FSPEC$:ON ERROR GOTO 0:END
JL 150 BEEP:COLOR 31:CLS:PRINT "Cannot read directory":COLOR 7:ON ERROR GOTO 0:END
```

This routine will ask you from which drive (A: or B:) you want to read the directory. If you have a single-drive system (drive A: only), remove the REM from line 100. Here's another routine that uses the second method:

```
KJ 1000 REM FSPEC$="A:*.*":GOTO 1040
LH 1010 PRINT:PRINT"Select drive: (";:COLOR 16,15:PRINT"A B";:COLOR 7,0:PRINT CHR$(29)CHR$(29)"/"CHR$(28)")"
HC 1020 DRIVE$=INKEY$+":":A=ASC(DRIVE$):IF (A OR 32)<97 OR (A OR 32)>98 THEN 1020
ME 1030 DRIVE$=CHR$(A AND 223)+":":FSPEC$=DRIVE$+"*.*"
LN 1040 DEF SEG=0:WIDTH 80
IF 1050 HEAD=1050:TAIL=1052:BUFFER=1054
OJ 1060 CLS:COLOR 23,0,0:PRINT"Reading disk directory"
DE 1070 COLOR 0:ON ERROR GOTO 1090
EJ 1080 FILES FSPEC$:ON ERROR GOTO 0:GOTO 1100
IL 1090 BEEP:COLOR 31:CLS:PRINT "Cannot read directory":COLOR 7:ON ERROR GOTO 0:END
MF 1100 DIM TT$(24):LOCATE 3,1:COLOR 7:ROWS=0
CD 1110 POKE HEAD,30:POKE TAIL,34:POKE BUFFER,0:POKE BUFFER+1,79:POKE BUFFER+2,13:POKE BUFFER+3,28:'Put code for End, Enter into keyboard
DE 1120 LINE INPUT TT$(ROWS):IF TT$(ROWS)<>"" THEN ROWS=ROWS+1:GOTO 1110
DN 1130 IF NOT DIMMED THEN DIM F$(ROWS*4-1):DIMMED=1
JP 1140 ROWS=ROWS-1:FOR I=0 TO ROWS:FOR J=0 TO 3
KE 1150 T$=MID$(TT$(I),J*18+1,12)
KD 1160 IF T$<>"" THEN F$(ENTRIES)=T$:ENTRIES=ENTRIES+1
KA 1170 NEXT J:NEXT I:ERASE TT$:ENTRIES=ENTRIES-1:DEF SEG:RETURN
```

This routine reads the filenames from the disk directory into an array named F$. One advantage of this method is that you need to look only once at the directory. Once the directory information is stored in a string, you can extract the filenames whenever it's convenient and print them in any format you like. With a little more programming, you could cursor through the directory to access a particular file, sort the directory entries alphabetically, catalog all your disks, or whatever. Again, remove the REM from line 1000 if you have a single-drive system.

64 RAM Report

Can you give me a short program that tests the RAM in my 64? I have had trouble running a particular BASIC program and think that my computer must have a defective RAM chip.

Fred Wayne

Though it's tempting to blame the hardware when things go awry, RAM chips rarely fail. Every time you turn on a Commodore 64, it performs a RAM verification as part of its normal power-up sequence. It tests every RAM address from location 1024 (the start of screen memory) upward until it hits a ROM (Read Only Memory) location that can't be written to. Unless a cartridge is installed, the test includes all of the BASIC programming space (locations 2048–40959).

Here's how the power-up test works. After saving the original contents of the tested memory location, the computer stores the value 85 ($55) there, then reads the contents back to make sure the operation was successful. Then it stores the value 170 ($AA) there and reads the contents again. To understand why those particular values are used, look at them in binary form:

01010101 = $55 = 85
10101010 = $AA = 170

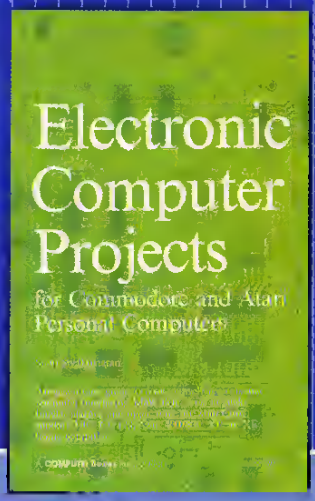
As you can see, every one bit in the first number is a zero bit in the second and vice versa. While you could test a location by successively writing and reading back every value from 0 to 255 (the maximum range for a single address), this method checks whether you can write and read back a one and a zero in each of the location's eight bits—which amounts to much the same thing. If a RAM address passes both tests, the 64 restores its original contents and proceeds to the next higher location, stopping as soon as it finds a read-back value that doesn't match what was just written. This normally happens at location 40960, the start of BASIC ROM. The location just below that (40959) is used as the top of BASIC memory.

Later in the startup sequence, the 64 subtracts 2048 from the top-of-memory value to calculate the number of bytes free for the startup message. Since 40959 − 2048 = 38911, the familiar message 38911 BASIC BYTES FREE tells you that the 64 just wrote and read back two values for every address in BASIC program space without detecting any errors.

If you're not convinced by the built-in test, here's a short ML program that tests the 64's RAM somewhat more thoroughly, writing and reading back every value from 0 to 255 before it concludes that a RAM address is functional. Be sure to save the program before you run it since

Two Exciting New Books

from COMPUTE!



COMPUTE!'s First Book of the Commodore 128

A spectacular collection of articles and programs exclusively for the Commodore 128 *in 128 mode.*

Edited

The editors at COMPUTE! Publications have collected some of the best games, programs, and tutorials for the Commodore 128 from *COMPUTE!* and *COMPUTE!'s Gazette*, plus some never-before-published articles and programs. Learn how to create windows, program sound, and make disks autoload. You'll even find a map of all the important memory locations. There's something for every 128 user. All programs run in 128 mode. *A disk is available which includes programs in the book, $12.95.*

$14.95 ISBN 0-87455-059-9

Electronic Computer Projects

Learn how to build all kinds of new devices to interface with your computer from inexpensive, available parts.

For the Commodore 64, 128, VIC, and any eight-bit Atari personal computer.

Soori Sivakumaran

This introduction to digital electronics and computer interfacing is the easy way to learn how computers interact with the outside world. Using a Commodore 64, 128, VIC, or any eight-bit Atari computer and *Electronic Computer Projects*, you'll be guided through the steps to building a joystick, light pen, game paddle, and numerous other devices. And since each project is independent from the others, you can choose only those projects that interest you. All the projects can be built at home and most require fewer than half a dozen parts.

$9.95 ISBN 0-87455-052-1

Visit your local book or computer store for these new titles. Or order directly from COMPUTE! Books. Call toll-free 800-346-6767 (in NY 212-887-8525) or write COMPUTE! Books, P.O. Box 5038, F.D.R. Station, New York, NY 10150.

Please include $2.00 per book ($5.00 air mail) for shipping and handling. NC residents add 4.5 percent sales tax. Allow 4–6 weeks from receipt of order for delivery.

COMPUTE! Publications, Inc. abc
Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies
825 7th Avenue, 6th Floor New York, NY 10019
Publishers of COMPUTE!, COMPUTE!'s Gazette, COMPUTE!'s Gazette Disk, COMPUTE! Books, and COMPUTE!'s Apple Applications

COMPUTE! books are available in the U.K., Europe, the Middle East, and Africa from Holt Saunders, Ltd., 1 St. Anne's Road, Eastbourne, East Sussex BN21 3UN, England, and in Canada from McGraw-Hill Ryerson Ltd., 330 Progress Ave., Scarborough, Ontario, Canada M1P 2Z5.

the ML portion erases the BASIC loader:

```
FK 10 ADR=49152
JG 20 READ BYT:IF BYT<>256 THE
      N POKE ADR,BYT:ADR=ADR+1
      :CK=CK+BYT:GOTO 20
RC 30 IF CK<>11516 THEN PRINT"
      ERROR IN DATA STATEMENTS
      --CHECK TYPING":END
JA 40 PRINT "PRESS RETURN TO C
      HECK BASIC RAM":PRINT
FB 50 PRINT "SYS 49152"CHR$(14
      5)CHR$(145)CHR$(145)
DR 49152 DATA 169,0,133,251,16
         9,8,133,252,32,228,25
         5
PD 4915B DATA 208,58,166,251,1
         65,252,32,205,189,169
         ,32
SQ 49164 DATA 32,210,255,160,0
         ,162,0,202,138,145,25
         1
FP 49170 DATA 209,251,240,18,1
         52,72,138,72,169,72,1
         60
JA 49176 DATA 192,32,30,171,10
         4,170,104,168,76,59,1
         92
DQ 49182 DATA 224,0,208,226,23
         0,251,208,2,230,252,1
         65
RQ 491BB DATA 252,201,160,208,
         193,96,157,95,18,66,6
         5
BX 49194 DATA 68,146,32,0,256
```

*This program checks the 51199 RAM locations from 2048 to 53247, which includes all of BASIC program space as well as the 8K of RAM underneath BASIC ROM and the 4K RAM zone from 49152 to 53247. If a location passes the test, its address is printed. If not, you'll see the message BAD in reverse video with an arrow pointing to the address. Since it performs over 13 million (51199*256) read/write operations, this program takes about 15 minutes to run. You can cut it short by pressing any key.*

Format With PRINT USING

I am having difficulty formatting an amortization table on my PCjr that will display dollars to two decimal places (to the cents place). Currently, my program drops the trailing zeros following a decimal point. Do you have a solution for this?

Keith Bovee

The answer is to substitute PRINT USING for the more common PRINT statement. PRINT USING is very versatile and can be used to format the output of string or numeric data. The general format for this command is:

PRINT USING *format$; expression(s)*

Replace format$ *with a string constant or variable containing special formatting characters (listed in your BASIC manual). The formatting characters tell the computer exactly how it should print the* expression *that follows the semicolon. The expression may be either string or*

numeric data, and you may include more than one expression.

Perhaps the most common use of PRINT USING is to format numeric data, a task that requires only two formatting characters. The number sign (#) reserves a digit position within the output string, and the dollar sign ($) stands for a dollar sign. For instance, type the following lines in direct mode (without line numbers):

```
X = 1234.00
PRINT X
PRINT USING "$####.##";X
```

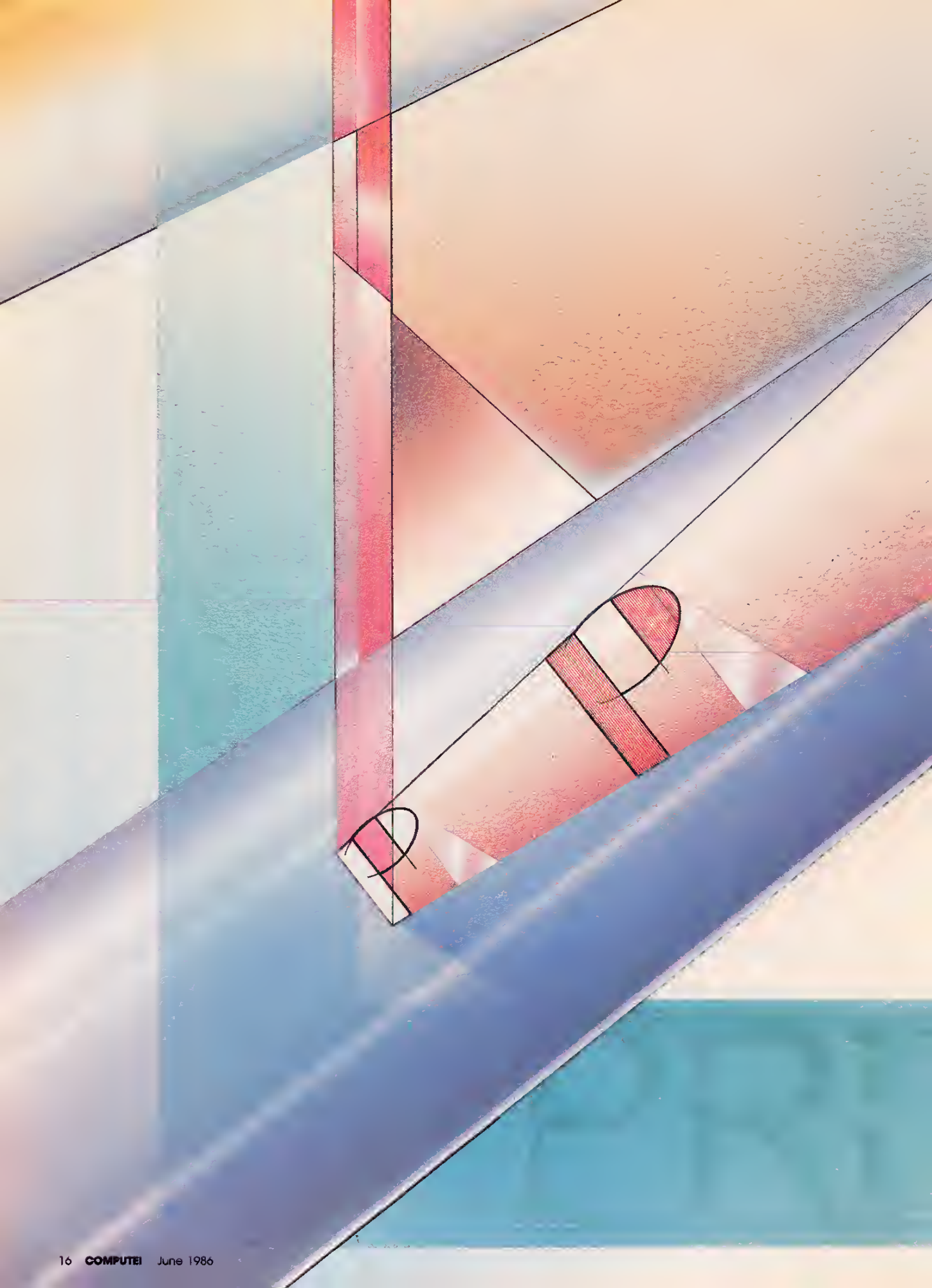
The first PRINT statement strips the decimal places, printing 1234. That's normal in BASIC, but undesirable in a program that requires dollars and cents format. The PRINT USING statement prints $1234.00 complete with a dollar sign and two decimal places. You can find additional examples of how to use PRINT USING in the IBM BASIC manual.

Disabling Atari BASIC

I recently purchased an Atari 800XL and some programs for it. One of the programs, *Micro League Baseball*, doesn't work because of the built-in BASIC. Is there an easy way to disable the computer's built-in BASIC temporarily?

Chris Greatens

To disable the built-in BASIC on an Atari 600XL, 800XL, or XE, hold down the OPTION key when turning on the computer. On an Atari 400, 800, or 1200XL, simply unplug the BASIC cartridge. ©



The Changing Face Of Printer Technology

Joan Rouleau, Research/Copy Editor

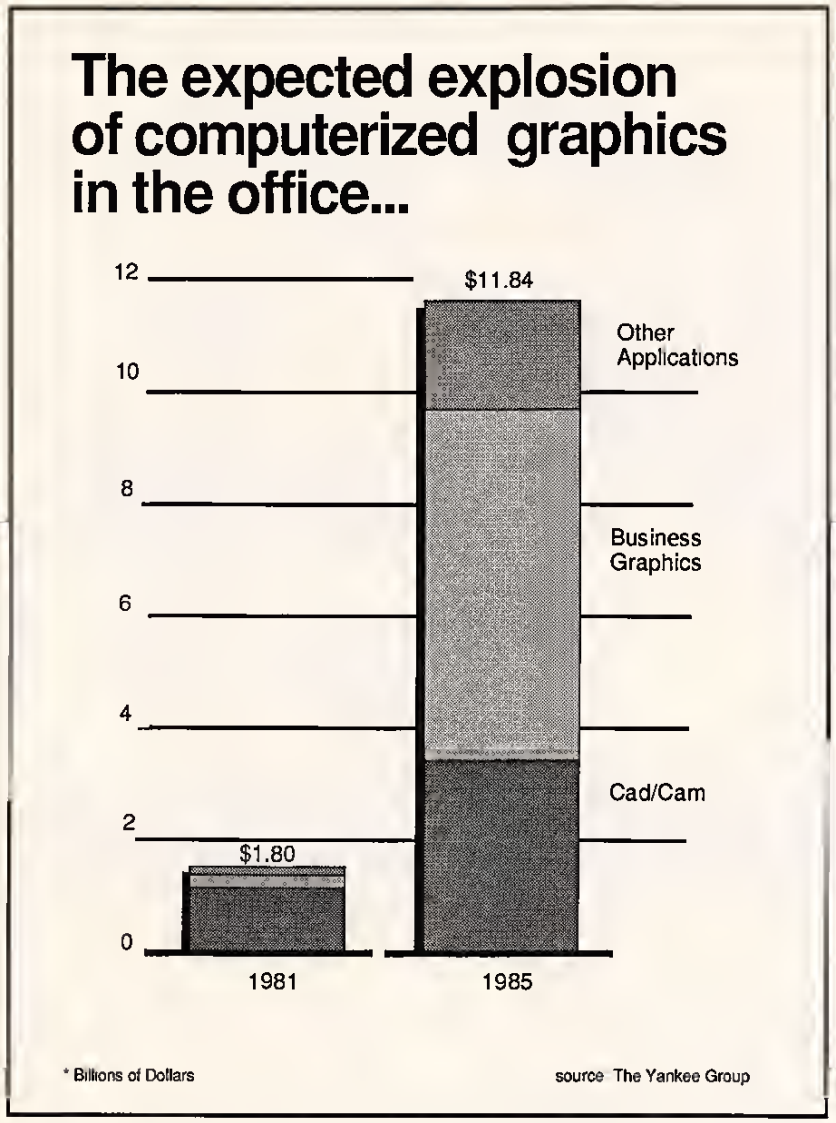
Today's printers are better than ever: They're faster, quieter, more versatile, less expensive, and produce higher-quality output than even their recent predecessors. Innovative new technologies—such as lasers and LED arrays—are offering more choices for home computer owners, while the more established technologies—such as dot-matrix, daisywheel, and ink-jet printers—have been greatly refined. Here's a look at some of the changes that are reshaping the printer marketplace.

Just five years ago, a 40-characters-per-second daisywheel printer was advertised in this magazine for almost $2,000—and that was a discount price. Today, that same cash can buy a silent, six-page-per-minute, multiple-font laser printer. Similarly, it wasn't very long ago that the blocky, awkward type produced by dot-matrix printers was appropriate only for printing draft copy. Now, with print resolution as great as 300 dots per inch in some new models, dot-matrix printers are reaching true letter quality. Better yet, the intense competition among manufacturers and retailers continues to push prices down and spawn a wider selection of features.

Printers are becoming such an integral part of the home computerist's workroom that only in a technical sense can they still be considered peripherals. The percentage of home computer owners with a printer nearly doubled between 1983 and 1985: from 28 percent to 53 percent, according to Link Resources, a New York market research firm. This trend is tied to an increase in word processing and business applications in the home, says the Link study.

Not only are more people buying printers, but they're also expecting more from the printers they buy. In particular, more and more people want better-quality print. A recent survey by another market research firm, Frost & Sullivan of New York, named print clarity as the single most important factor among those choosing a letter-quality printer. Other factors were ease of repair, long life, and then price.

Answering this demand for better print is a wide array of new nonimpact printers and substantial improvements in dot-matrix printers. Laser printers, once affordable only by large businesses, have just begun to drop in price. Within a couple of years, they, too, may become a contender in the home printer arena.



A sample of near-typeset quality output from a laser printer.

Laser printers were originally developed about a decade ago for use with mainframe computers, and they work much like photocopy machines. In a photocopier, the original is illuminated with a bright light that transfers the image of the page onto a light-sensitive drum. Through a thermal and electrochemical process, the drum then fuses the image onto another sheet of paper.

A laser printer works in a similar way, but uses a low-power laser to scribe the images onto the drum. Therefore, it is a *page printer*—it prints a whole page at a time rather than a single character at a time, like most printers. The newest laser printers can print up to a fleeting 12 pages per minute. And unlike most dot-matrix or daisywheel printers, they run quietly.

In the last couple of years, improvements in laser and photocopy technology brought the price of laser printers down to the $7,000 range, making them accessible to considerably more businesses. Then, last fall, QMS of Mobile, Alabama, introduced its Kiss laser printer for only $1,995, bringing this technology within reach of small businesses and some home users.

Among the other manufacturers who are developing laser printers in the $2,000 to $3,000 range are Okidata, Canon, Mannesmann Tally, Dataproducts, and ITT Qume. Many industry watchers predict that a $1,000 laser printer will be available by the end of 1987.

Others, however, are more skeptical about how soon the laser printer will become a mainstay in the home. Laser printers are still quite costly to manufacture, they argue, and it may be some time before these costs go down. Virtually all of the mechanisms for laser printers are made in Japan, and the devaluation of the dollar against the yen may keep laser printers more expensive for a while.

Perhaps in light of these considerations, some manufacturers are looking to other nonimpact technologies for their page printers. Particularly favored among several manufacturers is the light-emitting diode (LED) array. LEDs are tiny semiconductors that emit light when energized by a pulse of current, often seen as power indicator lights on stereos and computers. LED printers work something like laser printers, except they use an LED array to print the page instead of a laser. LED array printers are comparable in speed to laser printers, and because they have fewer moving parts, they are cheaper to produce and transport. Among the manufacturers who have chosen LED technology for their page printers are IBM and Datasouth.

Another nonimpact technology, *ion deposition*, also is making its debut. Instead of using light to transfer the image onto a drum, these printers shoot ion beams onto an electrically conductive drum.

$29.95. A SMALL PRICE TO PAY TO FIRE UP YOUR INVESTMENTS.

Invest it in the Dow Jones News/Retrieval® Membership Kit, and sample the nation's premier online financial information resource. With five free hours* to explore, a User's Guide to point the way, and a subscription to *Dowline*—the magazine of News/Retrieval, filled with case studies and insights—the Membership Kit will show you how to find fuel for your investment ideas.

Monitor current stock quotes, or track historical trends for companies or industries that interest you. Call up earnings and economic forecasts, and reports from major brokerage houses. Get breaking news as early as 90 seconds after it appears on the newswires, and news from *The Wall Street Journal* (online exclusively with News/Retrieval) back to January 1, 1984—to see what impact government or world events are having on the marketplace.

You can access News/Retrieval with almost any computer and modem, terminal or communicating word processor.

The Dow Jones News/Retrieval Membership Kit. Use it to kindle *your* success. Without getting burned.

To order or for more information, call **1-800-345-8500, Ext. 337.** (Alaska, Hawaii and foreign, call 1-215-789-7008, Ext. 337.) Or use the coupon today.

YES! I want my investments fired up!
Send me the Dow Jones News/Retrieval® Membership Kit.

☐ **Personal offer - $29.95 Each** includes • 5 Free Hours* • Single Password • User's Guide • 1 Year of *Dowline* • $12 Annual service fee waived for one year

☐ **Corporate offer** (for multiple users) - **$49.95 Each** includes • 8 Free Hours* • Multiple Passwords • User's Guide • 1 Year of *Dowline* • $12 Annual service fee waived for one year

☐ **Send more information**

$_____ ENCLOSED (Check or Money Order)

☐ Bill me later Charge to: ☐ AmExp ☐ MC ☐ VISA

Card No. _____ Exp Date_____

Signature _____

Name _____

Address _____

City/State/Zip _____

Daytime Phone Number _____

Computer Make & Model _____

Mail to: DJN/R Membership Kit, P.O. Box 300, Princeton, NJ 08543-0300
ATTN: John McGovern 5CO0506E

Dow Jones News/Retrieval®

Fuel for your ideas.

$29.95 Membership Kit also available at computer retailers and selected bookstores.

*Per account. Limited to new subscribers only. Free time (5 hours per Personal offer, 8 hours per Corporate offer) must be used within 30 days after receipt of password(s). Certain data bases have fees over and above usage charges which are excluded from free time offer.

Axiom's TX 2000 can print out a hard-copy of any video display.

Unlike laser or LED printers, ion deposition printers do not use heat as part of the transfer process. Companies investigating this technology include Star Micronics, Mannesmann Tally, and C. Itoh. Mannesmann's director of marketing, John Roberts, predicts that ion deposition printers are "the nonimpact technology that will probably come closest to replacing impact printers."

On another front, ink-jet printers continue to evolve and generate interest. These printers, as the name implies, spray a jet of ink from several tiny nozzles onto the paper. When first introduced, they could only produce draft-quality copy which had a tendency to smudge. Now major manufacturers such as Canon and Diablo are perfecting this technology and are reportedly developing high-resolution ink-jet printers.

Nonimpact technology is still in its infancy and will likely undergo a great many changes—in speed, price, and sophistication—over the next few years. "Lasers have opened the door. We're finding that there are other doors," says Tom Bongiorno, director of marketing for Star Micronics. "Just as when the first dot-matrix printer came out, it was certainly a breakthrough. Then print quality became better, prices dropped to one third or less than initially...the quality continues to pick up and prices will probably still drop."

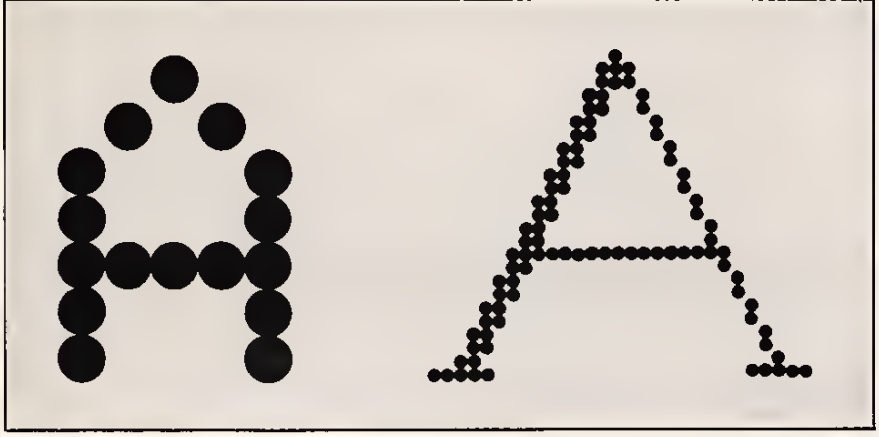
Does this surge of new nonimpact printers aimed at the home market mean the demise of dot-matrix? Not anytime soon. Dot-matrix printers are still considerably cheaper and have improved quite a lot over the past couple of years. Says Dennis Cox of Epson America, "There's continued optimism and growth in the dot-matrix industry. We're seeing more products become available, improved features, and new price levels."

Just in the last year, the resolution of dot-matrix print has greatly improved. All dot-matrix printers use a printhead which consists of a vertical row of stacked wires. As the printhead moves across the page, these wires are hammered onto the paper in different patterns to form characters in a rectangular matrix. When dot-matrix printers were first introduced, characters were formed in a 5 × 7 or 8 × 8 matrix (see the accompanying figure). Now several printers are on the market which have 18 or even 24 wires in their printheads. This allows the printer to form characters which are much better defined, and produce better graphics as well. Improvements have also been made which enable better positioning of the printhead, so even nine-wire printheads can produce higher-quality print than ever before.

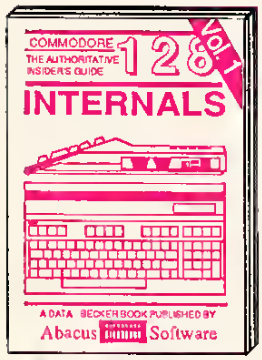
Among the new high-resolution dot-matrix printers are Toshiba's P321 ($699), which features a 24-wire printhead, 80-column carriage, and multiple type font

A 5 × 7 character from a 8-pin printhead.

A 24 × 30 character from a 24-pin printhead.



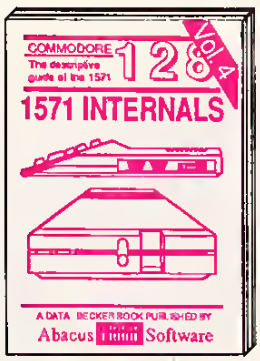
C-128™ and C-64™ REQUIRED READING



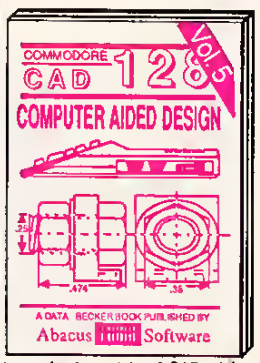
Detailed guide presents the 128's operating system, explains graphic chips, Memory Management Unit, 80 column graphics and commented ROM listings **500pp $19.95**

Get all the inside information on BASIC 7.0. This exhaustive handbook is complete with commented BASIC 7.0 ROM listings. Coming Summer '86 **$19.95**

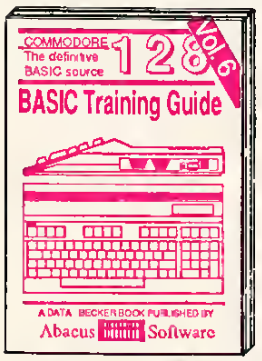
Filled with info for everyone. Covers 80 column hi-res graphics, windowing, memory layout, Kernal routines, sprites, software protection, autostarting **300pp $19.95**



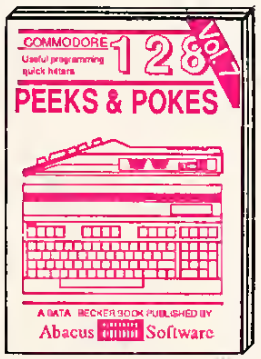
Insiders' guide for novice & advanced users. Covers sequential & relative files, & direct access commands. Describes DOS routines. Commented listings **$19.95**



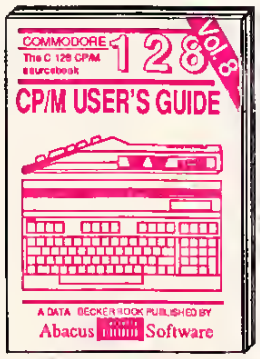
Learn fundamentals of CAD while developing your own system. Design objects on your screen to dump to a printer. Includes listings for '64 with Simon's Basic **300pp $19.95**



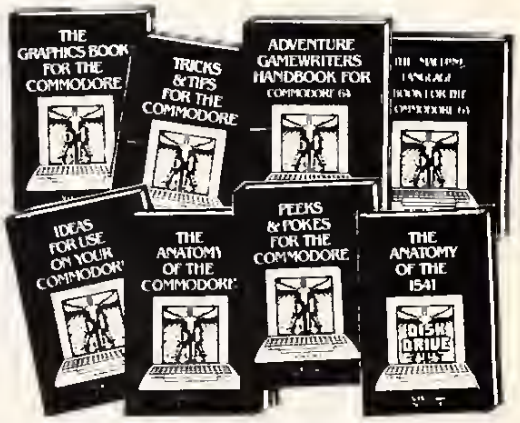
Introduction to programing, problem analysis, thorough description of all BASIC commands with hundreds of examples, monitor commands, utilities, much more. **$16.95**



Presents dozens of programming quick-hitters. Easy and useful techniques on the operating system, stacks, zero page, pointers, the BASIC interpreter and more. **$16.95**



Essential guide for everyone interested in CP/M on the 128. Simple explanation of the operating system, memory usage, CP/M utility programs, submit files & more **$19.95**



ANATOMY OF C-64 Insider's guide to the '64 internals. Graphics, sound, I/O, kernal, memory maps, more. Complete commented ROM listings. **300pp $19.95**

ANATOMY OF 1541 DRIVE Best handbook on floppy explains all. Many examples and utilities. Fully commented 1541 ROM listings. **500pp $19.95**

MACHINE LANGUAGE C-64 Learn 6510 code write fast programs. Many samples and listings for complete assembler, monitor, & simulator. **299pp $14.95**

GRAPHICS BOOK C-64 - best reference covers basic and advanced graphics. Sprites, animation, Hires, Multicolor, lightpen, 3D-graphics, IRQ, CAD, projections, curves, more. **359pp $19.95**

TRICKS & TIPS FOR C-64 Collection of easy-to-use techniques: advanced graphics, improved data input, enhanced BASIC, CP/M, more. **275pp $19.95**

1541 REPAIR & MAINTENANCE Handbook describes the disk drive hardware. Includes schematics and techniques to keep 1541 running. **200pp $19.95**

ADVANCED MACHINE LANGUAGE Not covered elsewhere. - video controller, interrupts, timers, clocks, I/O, real time, extended BASIC, more. **219pp $14.95**

PRINTER BOOK C-64/VIC-20 Understand Commodore, Epson-compatible printers and 1520 plotter. Packed. utilities, graphics dump; 3D plot; commented MPS801 ROM listings, more. **330pp $19.95**

SCIENCE/ENGINEERING ON C-64 In depth intro to computers in science. Topics: chemistry, physics, biology, astronomy, electronics, others. **350pp $19.95**

CASSETTE BOOK C-64/VIC-20 Comprehensive guide; many sample programs. High speed operating system, fast file loading and saving. **225pp $14.95**

IDEAS FOR USE ON C-64 Themes: auto expenses, calculator, recipe file, stock lists, diet planner, window advertising, others. Includes listings. **200pp $12.95**

COMPILER BOOK C-64/C-128 All you need to know about compilers: how they work, designing and writing your own, generating machine code. With working example compiler. **300pp $19.95**

Adventure Gamewriter's Handbook Step-by-step guide to designing and writing your own adventure games. With automated adventure game generator. **299pp $14.95**

PEEKS & POKES FOR THE C-64 Includes in-depth explanations of PEEK, POKE, USR, and other BASIC commands. Learn the "inside" tricks to get the most out of your '64. **200pp $14.95**

Optional Diskettes for books
For your convenience, the programs contained in each of our books are available on diskette to save you time entering them from your keyboard. Specify name of book when ordering. **$14.95 each**

C-128 and C-64 are trademarks of Commodore Business Machines Inc

Abacus Software

P.O. Box 7219 Grand Rapids, MI 49510 - Telex 709-101 - Phone (616) 241-5510

Call **now** for the name of your nearest dealer. Or to order directly by credit card, MC, AMEX of VISA call **(616) 241-5510**. Other software and books are available–Call and ask for your **free** catalog. Add $4.00 for shipping per order. Foreign orders add $10.00 per book. Dealer inquires welcome–1400+ nationwide.

The first laser printer for under $2,000: The new Kiss from QMS.

cartridges. Okidata's 2410 ($2,395) also has a 24-wire printhead and can print 136 columns across. Star Micronic's NB-15 ($1,449), another 24-wire printer, produces letter quality at 100 characters per second (cps) and draft quality at 300 cps, and has a 16K buffer. NEC has a new line of dot-matrix printers which includes the 24-wire Pinwriter P5 ($1440–$1560). Fujitsu America's 24-wire dot-matrix printer has a liquid crystal display which shows the print status. Recently released 18-wire dot matrix printers include Mannesmann Tally's MT490 and Datasouth's DS440 (prices haven't been released for either machine). While these may still be a little too expensive for many home users, the prognosis for less costly high-resolution dot-matrix printers over the next few years is very good.

Dot-matrix printers are also improving in speed and other features. "What we feel is the trend for the dot-matrix market right now is that more and more features are being built into printers," says Frank Rexach, product manager for C. Itoh. Some of the features in C. Itoh's new C310 are 300 cps draft printing, paper feeding from the top, bottom, and rear, and all control keys located on the front panel.

The dot-matrix printer's chief rival, the daisywheel, stands a chance of being superseded by the letter-quality dot-matrix printers and the new nonimpact devices. Daisywheel printers work much like typewriters and used to be the only way to get letter-quality print. Now many manufacturers have slowed or stopped their production of daisywheels while expanding into the dot-matrix and nonimpact areas.

As Mannesmann Tally's John Roberts says, "Daisywheel manufacturers are the most subject to displacement by the laser printer." Or, as another manufacturer puts it, "I wouldn't want to be *only* in the daisywheel market right now."

Anyone who sever tried to use a printer for graphics knows how difficult it can be. While virtually all dot-matrix printers have some graphics capability, there are no standard control codes for accessing this feature. Programs that print graphics have a hard time supporting all the different printers that are available.

This situation has led to the development of *page description languages*. With a page description language, your software can access features like graphics and text in several fonts without knowing what kind of printer you have. All that's necessary is that your printer understand the page description language that your software is generating. One of the most popular of these is *PostScript* by Adobe Systems, which can be used with the Apple LaserWriter and other printers.

Thanks to *PostScript*, high-quality printing is available to those who can't justify the expense of a laser printer for occasional printing jobs. A document description can be sent over a phone line with a modem, and some professional typesetting machines understand *PostScript*. So in some areas, it's already possible to create and lay out a document, upload it to a print shop that has one of these typesetters, and have it typeset without leaving your home.

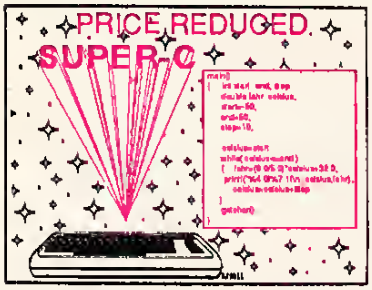
Other new developments in the area of printer graphics include the *digital videoprinter*—a printer which makes hardcopy from any type of raster-scan video display (including computer monitors and TVs). The TX 2000 videoprinter, recently released by Axiom Corporation, can capture a moving video image and even rotate or reverse the image. The TX 2000 lists for $1,995.

A number of new color printers are also opening up new graphics possibilities for home computers. Juki has just released a nine-wire color dot-matrix printer which can produce up to seven colors from a four-color ribbon. Fujitsu America is offering a color version of its 24-wire dot-matrix printer. And a few companies are developing color laser printers for use with personal computers. It'll be a while before these devices are found in many homes, though—color laser printers, like color photocopiers, are still very expensive.

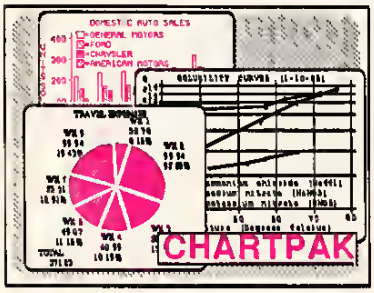
Advances in printer technology continue at a rapid pace, and the printers that have been recently announced demonstrate just how quickly the market is progressing. Whether you need a simple and inexpensive dot-matrix printer for casual use or a state-of-the-art laser printer for near-typeset quality documents, the latest printers provide unprecedented performance at far lower prices. ©

'128™ and C-64™ SPECTACULAR SOFTWARE

The *complete* compiler and development package. Speed up your programs 5x to 35x. Many options: flexible memory management; choice of compiling to machine code, compact p-code or both. '128 version: 40 or 80 column monitor output and FAST-mode operation. '128 Compiler's extensive 80-page programmer's guide covers compiler directives and options, two levels of optimization, memory usage, I/O handling, 80 column hi-res graphics, faster, higher precision math functions, speed and space saving tips, more. A great package that no software library should be without. **128 Compiler $59.95**
64 Compiler $39.95



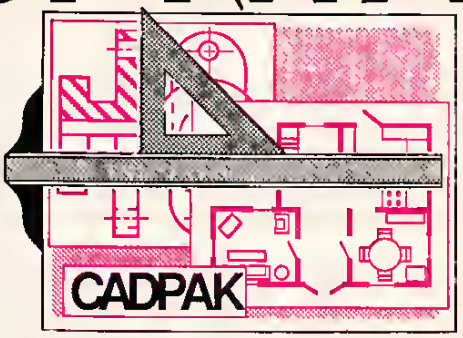
For school or software development. Learn C on your Commodore with our in-depth tutorial. Compile C programs into *fast* machine language. C-128 version has added features: Unix™-like operating system; 60K RAM disk for fast editing and compiling Linker combines up to 10 modules; Combine M/L and C using CALL; 51K available for object code; Fast loading (8 sec. 1571, 18 sec. 1541); Two standard I/O librarys plus two additional libraries—math functions (sin, cos, sqrt, etc.) & 20+ graphic commands (line, fill, dot, etc.).
C-128 $79.95
C-64 $79.95



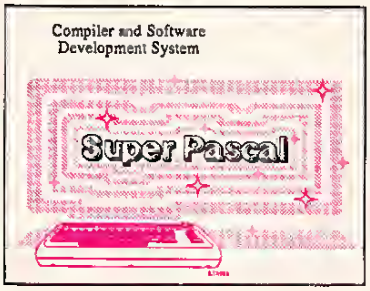
Easily create professional high quality charts and graphs without programming. You can immediately change the scaling, labeling, axis, bar- filling, etc. to suit your needs. Accepts data from CalcResult and MultiPlan. C-128 version has 3X the resolution of the '64 version. Outputs to most printers.
C-128 $39.95
C-64 $39.95

PowerPlan
One of the most powerful spreadsheets with integrated graphics. Includes menu or keyword selections, online help screens, field protection, windowing, trig functions and more. PowerGraph, the graphics package, is included to create integrated graphs & charts. **C-64 $39.95**

COBOL Compiler for the C-64	**$39.95**
Ada Compiler for the C-64	**$39.95**
VidaoBasic Language for the C-64	**$39.95**



Remarkably easy-to-use interactive drawing package for accurate graphic designs. New dimensioning features to create exact scaled output to all major dot-matrix printers. Enhanced version allows you to input via keyboard or high quality lightpen. Two graphic screens for COPYing from one to the other. DRAW, LINE, BOX, CIRCLE, ARC, ELLIPSE available. FILL objects with preselected PATTERNS; add TEXT; SAVE and RECALL designs to/from disk. Define your own library of symbols/objects with the easy-to-use OBJECT MANAGEMENT SYSTEM–store up to 104 separate objects.
C-128 $59.95
C-64 $39.95



Not just a compiler, but a complete system for developing applications in Pascal with graphics and sound features. Extensive editor with search, replace, auto, renumber, etc. Standard J & W compiler that generates fast machine code. If you want to learn Pascal or to develop software using the best tools available–**SUPER Pascal** is your first choice.
C-128 $59.95
C-64 $59.95

OTHER TITLES AVAILABLE:

Technical Analysis System
Sophisticated charting and technical analysis system for serious investors. Charting and analyzing past history of a stock, TAS can help pinpoint trends & patterns and predict a stock's future. Enter data from the keyboard or from online financial services. **C-64 $59.95**

Personal Portfolio Manager
Complete portfolio management system for the individual or professional investor. Easily manage your portfolios, obtain up-to-the-minute quotes and news, and perform selected analysis. Enter quotes manually or automatically through Warner Computer Systems. **C-64 $39.95**

Xper
XPER is the first "*expert system*" for the C-128 and C-64. While ordinary data base systems are good for reproducing facts, XPER can derive knowledge from a mountain of facts and help you make expert decisions. Large capacity. Complete with editing and reporting. **C-64 $59.95**

Abacus Software

P.O. Box 7219 Grand Rapids, MI 49510 - Telex 709-101 - Phone (616) 241-5510

Call **now** for the name of your nearest dealer. Or to order directly by credit card, MC, AMEX of VISA call **(616) 241-5510**. Other software and books are available–Call and ask for your **free** catalog. Add $4.00 for shipping per order. Foreign orders add $12.00 per item. Dealer inquires welcome–1400+ nationwide.

Buyer's Guide To The PRINTERS OF 1986

If you already know how you'll be using a printer and what features you'll need before you start shopping, the hard part is over. There are many good printers available for a variety of applications, and prices continue to drop as manufacturers expand their hardware lines.

We've gathered information on printers in the under-$800 price range and listed some of the most important features in the following chart. Any omissions are not an editorial judgment of quality.

Here's a brief explanation of the major categories on the chart:

Compatibility. Chances are your computer has either a serial or parallel port (or both) that hooks up to a printer. Some printers come in either serial or parallel versions; some offer both interfaces; and some are available in parallel or serial only. If the printer you want comes only in a version that doesn't support your computer, you should be able to buy a separate interface that allows that configuration. Also, many printer manufacturers sell interfaces designed specifically for certain computers, avoiding any compatibility problems.

Be careful here. In some situations, a particular interface will let you print text, but will be incapable of producing graphics. If there's any doubt, it's best to try and test your setup at a computer dealer.

Print technology. This refers to how characters and graphics are actually transferred from printer to paper. There are three types in this price range: impact, thermal, and ink-jet.

Impact printers form characters by striking the paper through an inked ribbon, either with a *daisy-wheel* (a small wheel whose spokes have letters and numbers on their tips), or with a printhead containing a column of tiny wires or pins that form characters and graphics *(dot-matrix). Thermal* printers use either a column of hot pads that change the color of heat-sensitive paper, or a column of tiny spark plugs that evaporate a special aluminum coating onto the paper, exposing an underlying dark surface. Thermal printers require special paper, which often costs more than regular paper and has a shorter life. *Thermal transfer* printers work with any kind of paper because they use ribbons; heat from the printhead melts a waxlike ink onto the paper. *Ink-jet* printers spray ink onto the paper through tiny holes.

Speed. How fast does the printer operate? This can vary if the printer offers different modes. *Draft mode* is usually the fastest, but produces rougher, fainter type. *Near letter quality (NLQ),* or *correspondence mode,* takes longer to print, but looks more polished. Some printer speeds vary depending on the type of *font* (for example, pica or elite) used. In our chart, a wide speed range, like 30–120 characters per second (cps), indicates that the printer offers some kind of correspondence-quality type.

Pitch. This indicates how many characters fit on a line, measured in characters per inch (cpi) or characters per line (cpl). The pitch range for a printer often varies greatly, especially if it is capable of printing several types of fonts.

Buffer. A buffer is an area of memory in a printer that can store a fixed amount of text while the printer is working, freeing up the computer for other tasks. Most printers in the under-$800 price range still have rather small buffers, so if you'll be doing many long printing jobs, you may want to consider buying an add-on buffer.

Feed type. *Friction-feed* printers grip the paper and move it around the platen much as a typewriter does, while *tractor-feed* printers have teeth at both ends of the platen that grab holes at the edges of continuous-feed paper. Many printers have optional tractors.

Suggested retail price. This is the price set by the manufacturer; you may well find it at a lower price if you shop around.

A full explanation of the graphics capabilities of each printer takes more space than we have available. If you plan to use your printer extensively for printing graphics, make sure it's capable of doing what you need before you buy.

Model Name	Manufacturer/ Distributor	Compatibility	Print Technology	Speed	Pitch	Buffer	Feed Type	Warranty	Suggested Retail Price	Comments
Alphacom Aero	Alphacom, Inc.	Parallel or serial std	Dot matrix	130 cps	5-16.5 cpi	2K	Friction and pin std	6 months	299	Dot addressable and fully programmable graphics
Alphapro 101	Alphacom, Inc.	Parallel or serial std	Daisy wheel	20 cps	10–15 cpi	93 characters	Friction std	6 months	399.95	
Image Writer	Apple Computer	Serial std	Dot matrix	120 cps	4.5–17 cpi	32K available	Friction or tractor std	90 days	595	
Image Writer II	Apple Computer	Compatible with Apple II, III, Lisa	Dot matrix	180–250 cps	4.5–17 cpi	32K available	Friction and adjustable width pin std; sheet feeder opt	90 days	595	
Scribe	Apple Computer	Serial std	Dot matrix	50–80 cps	10–17 cpi	N/A	Tractor std	90 days	249	
Aprotek Daisy 1120	Aprotek	Parallel std; serial opt	Daisy wheel	20 cps	10–15 cpi and proportional	2K	Friction std; tractor and cut sheet feeder opt	1 year	299.95	Two week trial available
Elite 5	Aprotek	Parallel and Commodore serial std	Daisy wheel	10–12 cps	10 cpi	None	Friction std; tractor opt	1 year	174.95	Two week trial available;Commodore version $184.95
SP-1000	Aprotek	Parallel or direct connect, IBM standard	Dot matrix	20–70 cps	10-15 cpi	1.5K	Friction and tractor std	2 years	239.95	Dot addressable graphics Commodore graphics built in on Commodore version ($219.95)
DX 1500	Axiom Corp	Parallel std	Daisy wheel	14 cps	10–12 cpi and proportional	2K	Friction std; tractor and auto cut sheet feeder opt	1 year	349	
DX 2000	Axiom Corp	Parallel std	Daisy wheel	20 cps	10–15 cpi and proportional	2K	Friction std; tractor opt	1 year	449	
DX 2500	Axiom Corp	Parallel std; serial opt	Daisy wheel	20 cps	10–15 cpi and proportional	2K	Friction std; tractor opt	1 year	499	
DX 3500	Axiom Corporation	Parallel and serial std	Daisy wheel	35 cps	10–15 cpi and proportional	2K	Friction std; tractor opt	1 year	699	
Thin Print 80P/80S	Axonix Corp	Parallel or serial std	Thermal transfer dot matrix	40 cps	10–17 cpi	2K	Friction std	90 days	339	High resolution graphics; portable (battery powered); AC adaptor included
Thin Print 100	Axonix Corp	Parallel or serial std	Thermal transfer dot matrix	25-100 cps	10-17 cpi	2K	Friction std	90 days	299	
Thin Write 80W	Axonix Corp	Parallel and serial std; HP-1L opt	Dot matrix	24–100 cps	5–17 cpi	2K	Friction std; tractor opt	90 days	449	Battery powered
Thin Write 100	Axonix Corp	Parallel and serial std	Dot matrix	25-100 cps	5-17 cpi	4K	Friction and pin std; tractor opt	90 days	479	
Blue Chip 120/NLQ	Blue Chip Electronics	Parallel std	Dot matrix	120 cps	5–17 cpi	3 lines	Tractor std	6 months	279	
D12/10	Blue Chip Electronics	Commodore serial std	Daisy wheel	12 cps	10 cpi	2K	Friction std; tractor opt	6 months	249	Comes with Fleetwriter III wordprocessor
D20/10	Blue Chip Electronics	Parallel and Commodore serial std	Daisy wheel	20 cps	10 cpi	2K	Friction std; tractor opt	6 months	279	Comes with Fleetwriter III wordprocessor
M 120/10	Blue Chip Electronics	Parallel std; serial opt	Dot matrix	120 cps	5–17 cpi	3 lines (4K opt)	Friction and tractor std	6 months	229	Dot addressable graphics
M 150/15	Blue Chip Electronics	Parallel std; serial opt	Dot matrix	130 cps	5–17 cpi	2K	Friction and tractor std	6 months	349	Dot addressable graphics
HR-10	Brother International Corp.	Parallel and serial std	Daisy wheel	12 cps	10–15 cpi and proportional	2K	Friction and tractor std	90 days	349	
M-1109	Brother International Corp	Parallel and serial std	Dot matrix	25-100 cps	10 cpi	2K	Friction std; tractor opt	1 year	269	
M-1509	Brother International Corp	Parallel and serial std	Dot matrix	45–180 cps	10 cpi	3K	Friction and tractor std; cut sheet feeder opt	1 year	549	Seven bit image graphics
Prowriter Jr	C. Itoh	Parallel std; serial opt	Dot matrix	20–105 cps	10–17 cpi	1 line	Friction and tractor std	1 year	329	
Prowriter 8510 S series	C. Itoh	Serial or parallel std	Dot matrix	45–180 cps	10–17 cpi	2K	Tractor and friction std	1 year	499	
Prowriter C-310	C. Itoh	Parallel std; serial opt	Dot matrix	28–300 cps	10–15 cpi	2K	Friction and tractor std	1 year	599	
Y10-20	C. Itoh	Serial or parallel std	Daisy wheel	22 cps	10–15 cpi	2K	Friction std; tractor opt	1 year	549	
Legend 808	CAL-ABCO	Parallel std	Dot matrix	50–100 cps	5–17 cpi	1 line	Friction and tractor std	90 days	199	Bit image graphics
Legend 1080A	CAL-ABCO	Parallel std; serial opt	Dot matrix	70-140 cps	5-17 cpi	1 line std (4K opt)	Friction and tractor std	90 days	349	Five switch selectable modes including NLQ
Legend 1380	CAL-ABCO	Parallel std; serial opt	Dot matrix	160 cps	5-17 cpi	2K std; 4K opt	Friction and tractor std	90 days	379	10" version of 1385
Legend 1385	CAL-ABCO	Parallel std; serial opt	Dot matrix	160 cps	5–17 cpi	2K (4K opt)	Friction and tractor std	90 days	449	
A-40	Canon USA, Inc.	Parallel std	Dot matrix	27–140 cps	10 cpi std	1 4K	Friction and tractor std	1 year	349	
A-50	Canon USA, Inc	Parallel std; serial opt	Dot matrix	34–180 cps	10 cpi std	2K	Friction and tractor std; forms guide opt	1 year	499	Epson FX compatible
A-55	Canon USA, Inc	Parallel std	Dot matrix	27–140 cps	10 cpi std	1 4K	Friction and tractor std; forms guide opt	1 year	699	

COMPUTER MAIL ORDER

THE CMO ADVANTAGE

TO ORDER CALL TOLL FREE
1-800-233-8950
DEPARTMENT A206

OR MAIL YOUR ORDER TO:

COMPUTER MAIL ORDER
Department A206
477 E. Third Street
Williamsport, PA 17701

VISA MasterCard DINERS CLUB

POLICY

Add 3% (Minimum $7.00) shipping and handling. Larger shipments may require additional charges. Personal and company checks require 3 weeks to clear. For faster delivery use your credit card or send cashier's check or bank money order. Pennsylvania residents add 6% sales tax. All prices are subject to change and all items are subject to availability. Defective software will be replaced with the same item only. Hardware will be repaired or replaced at our discretion within the the terms and limits of the manufacturer's warranty. We cannot guarantee compatibility. All sales are final and returned shipments are subject to a restocking fee.

EDUCATIONAL INSTITUTIONS
CALL TOLL FREE
1-800-221-4283
CUSTOMER SERVICE
& TECHNICAL SUPPORT
1-717-327-1450

CANADIAN ORDERS
1-800-268-3974
Ontario/Quebec
1-416-828-0866
In Toronto
1-800-268-4559
Other Provinces
TELEX: 06-218960
2505 Dunwin Drive,
Mississauga, Ontario
Canada L5L1T1

All prices shown are for U.S.A. orders. Call the Canadian Office for Canadian prices.

THE CMO ADVANTAGE

- Next day shipping on all in-stock items
- Free easy access order inquiry
- Orders from outside Pennsylvania save state sales tax
- Free technical support from our factory trained technicians
- There is no limit and no deposit on C.O.D. orders
- There is no extra charge for using your Visa or MasterCard and your card is not charged until we ship
- No waiting period for cashier's checks
- We accept purchase orders from qualified corporations. Subject to approval.
- Educational discounts available to qualified institutions. (See the toll free educational phone number above.)
- FREE CATALOG MEMBERSHIP

HOME COMPUTERS

APPLE

APPLE IIe	CALL
APPLE IIc	CALL
IIc LCD Display	CALL

ATARI

65XE (64K)	**CALL**
130XE (128K)	**CALL**
520ST (512K)	**CALL**
800XL 64K	CALL
1010 Recorder	$49.99
1050 Disk Drive	CALL
1020 Printer	$29.99
1027 Letter Quality Printer	$129.00
1030 Direct Connect Modem	$59.99
Comrex 220 Atari	$89.99

COMMODORE

Amiga Package	
512K, 2 Drive, RGB Monitor	**$1799.00**
C64 Package	
C64, C1541, C1802	**$559.00**
C128 Computer	**$269.00**
C1571 (Disk Drive for C128)	**$249.00**
C1902 (RGB 13" Monitor for C128.	**CALL**
C1670 (Modem for C128)	**$179.00**
C1530 Datasette	$39.99
C1660 Auto Modem	$59.99
DPS 1101 Daisy Printer	$339.00
Comrex 220 (C64 Interface)	$89.99
Xetec SuperGraphix 8K	$69.99

GRAPHICS

Polaroid

Palette	$1399.00
Power Processor	$229.00
Illuminated Slide Mounter	$39.99
Polacolor 2 Pack film	$18.99

ACCESSORIES

CURTIS POWER STRIPS

Diamond SP-1	$29.99
Emerald SP-2	$39.99
Sepphire SPF-1	$49.99
Ruby SPF-2	$59.99
Safe Strip SP-3	$19.99

KENSINGTON

MasterPiece	$99.99
MasterPiece Plus	$119.00

HOME SOFTWARE

ARTWORX

Bridger 4.0	$17.99
Strip Poker	$19.99

BROOERBUND

Print Shop (All)	$29.99
Graphics Library I, II, III (All)	$17.99
Bank Street Writer (64, AT)	$34.99
Bank Street Writer (IIe/IIc)	$44.99

BATTERIES INCLUDED

Paperclip (AT, 64/128)	$37.99
Paperclip (IIe/IIc)	$44.99
Homepack (64/AT)	$29.99
D.E.G.A.S. (520)	$27.99

CONTINENTAL SOFTWARE

Home Accountant (All)	$44.99
Tax Advantage (All)	$44.99
Home Acct./Tax Bundle (All)	$59.99

JANE

Wordpro/dBase (C64)	$39.99

MICROPROSE

Kennedy Approach (All)	$24.99
Acrojet (All)	$24.99
Silent Service (All)	$24.99

MICRO SYSTEMS

Analyze for Amiga	$59.99

PFS

File (64)	$39.99
File, Report, Graph (IIe/IIc)	$79.99
Write w/Proof (IIe/IIc)	$79.99
Workmates (IIe/IIc)	$169.00

PSI

Fleet Systems III (C128)	$59.99

ROCKLAN
Game Carts for Atari

Gorf	$3.99
Wizard of Wor	$3.99
Space Journey	$3.99
Journey to Planet	$3.99
Rack-Em-Up	$3.99
Diamond Mind	$3.99

SPRINGBOARD

Newsroom (64)	$32.99
Clip Arts (64)	$19.99
Newsroom (Apple)	$39.99
Clip Arts (Apple)	$22.99

SUBLOGIC

Flight Simulator (All)	$37.99
Jet (84)	$37.99

MODEMS

ANCHOR

Volksmodem	$59.99
Volksmodem 300/1200	$189.99
Signalman Express	$259.00
Lightning 2400 Baud	$399.00
Expressi (PC Halfcard)	$189.00
6470 (64/128) 300/1200 Baud	$139.00

AST

Reach 1200 Baud Half Card	$399.00

DIGITAL DEVICES

AT300 - 300 Baud (Atari)	$99.99

EVEREX

1200 Baud Internal (IBM/PC)	$179.00

Hayes

Smartmodem 300	$139.00
Smartmodem 1200	$389.00
Smartmodem 1200B	$359.00
Smartmodem 2400	$599.00
Micromodem IIe	$149.00
Smart Com II	$89.99
Chronograph	$199.00
Transet 1000	$309.00

Novation

Smart Cat Plus	$299.00
J-Cat	$99.99
Novation 2400	$499.00
Apple Cat II	$219.00
212 Apple Cat II	$379.00
Apple Cat 212 Upgrade	$229.00

QUADRAM

Quadmodem II	
300/1200	$339.00
300/1200/2400	$499.00

SUPRA

MPP-1064 AD/AA (C-64)	$69.99

PORTABLE COMPUTERS

HEWLETT PACKARD

41CV	$139.00
41CX	$199.00
HP 11C	$49.99
HP 12C	$75.99
HP 15C	$75.99
HP 16C	$89.99
HPIL Module	$98.99
HPIL Cassette or Printer	$359.99
Card Reader	$143.99
Extended Function Module	$63.99
Time Module	$63.99

We stock the full line of HP calculator products

NEC

PC-8401 LS	$699.00
PC-8201 Portable Computer	$339.00
PC-8231 Disk Drive	$599.00
PC-8221A Thermal Printers	$149.00
PC-8281A Date Recorder	$99.99
PC-8201-06 8K RAM	$79.99

SHARP

PC-1350	$149.00
PC-1261	$149.00
PC-1500A	$169.00
PC-1250A	$89.99
CE-125 Printer/Cassette	$129.00
CE-150 Color Printer Cassette	$149.00
CE-161 16K RAM	$129.00

DISKETTES

Dennison

Elephant 5¼" SS/SD	$9.99
Elephant 5¼" SS/DD	$11.99
Elephant 5¼" DS/DD	$14.99
Elephant Premium DS/DD(50)	$69.99
Elephant 3½" SS/DD	$24.99

IBM

5¼" DS/DD floppy disks
(Box of 10) $26.99

maxell

3½" SS/DD (10)	$18.99
3½" DS/DD (10)	$29.99
5¼" MD-1 SS/SD (10)	$11.99
5¼" MD-2 DS/DD (10)	$16.99
5¼" MD-2 HD for AT (10)	$32.99
3½" 5 pack SS/DD/Case	$9.99

Verbatim.

5¼" SS/DD	$12.99
5¼" DS/DD	$24.99
Disk Analyzer	$24.99

DISK HOLDERS

AMARAY

50 Disk Tub 5¼"	$9.99
30 Disk Tub 3½"	$9.99
80 Column Printer Stand	$14.99

INNOVATIVE CONCEPTS

Flip'n File 10	$2.49
Flip'n File 50	$14.99
Flip'n File 50 w/lock	$19.99
Flip'n File Data Case	$9.99

DRIVES

HARD

CORE

AT20-AT72MB	CALL

EVEREX

60 Meg Internal Backup System	$799.00

IOMEGA

A110H Single 10	CALL
A210H 10 + 10	CALL
A120H Single 20	CALL
A220H 20 + 20	CALL
Save on 10 & 20 Carts	CALL

IRWIN

Tape Backup	CALL

KITS

10 Meg with controller	$399.00
20 Meg with controller	$499.00

PRIAM

40, 60 MB Inner Space	CALL
Shared Data	CALL
Shared Space	CALL

TALLGRASS

25, 35, 50, 80 meg (PC)	from $1299.00

FLOPPY

ALLIED TECHNOLOGY

Apple II,II+,IIe ½ height	$109.00

INDUS

Atari GT	$199.00
C-64 /128 GT	$199.00

MSD

SD1 C-64 Single	$219.00
SD2 C-64 Dual	$469.00

TANOON

320K 5¼" (PC)	$119.00

TEAC

320K 5¼"	$119.00

DEPT. A206 CALL TOLL-FREE

COMPUTER MAIL ORDER

MONITORS

AMDEK

Video 300 Green	$119.00
Video 300A Amber	$129.00
Video 310A Amber TTL	$159.00
Color 600 Hi-Res RGB	$399.00
Color 722 Dual Mode	$529.00
Color 725	CALL
Color 730	CALL

MAGNAVOX

8562 RGB/Composite	$279.00
613 TTL Green	$99.99
623 TTL Amber	$99.99

NEC

JB1205A	$79.99
JB1270G/1275A (ea.)	$99.99
JB1260G TTL Green	$129.00
JB1285A TTL Amber	$129.00
JC1401 Multi Sync RGB	CALL

PRINCETON

MAX-12 Amber	$179.00
HX-9 9" RGB	$469.00
HX-9E Enhanced	$519.00
HX-12 12" RGB	$469.00
HX-12E Enhanced	$559.00
SR-12 Hi-Res	$599.00
SR-12P Professional	$699.00

QUADRAM

8400 Quadchrome I	$499.00
8410 Quadchrome II	$339.00
8420 Amberchrome	$179.00
8500 Quad Screen	$1449.00

TAXAN

115 12" Green	$119.00
116 12" Amber	$129.00
121 TTL Green	$139.00
122 TTL Amber	$149.00
220 14" Color Composite	$179.00
620 640x200 RGB	$439.00
630 640x200 RGB	$489.00
640 720x400 RGB	$539.00

ZENITH

ZVM 1220 Amber	$99.99
ZVM 1230 Green	$99.99
ZVM 1240 IBM Amber	$149.00
ZVM 135 RGB	$459.00
ZVM 1330 RGB	$459.00
ZVM 1360 RGB	CALL
ZVM 1380 E G Comp	CALL

INTERFACES

AST

Multi I/Q (Apple II)	$149.00

DIGITAL DEVICES

Ape Face (Atari)	$49.99
U-Print A (Atari)	$54.99
U-A16/Buffer (Atari)	$74.99
U-Call Interface (Atari)	$39.99
U-Print C (C64)	$49.99
P-16 Print Buffer	$74.99
U-Print 16 apple IIc	$89.99

MICRO R & O

Apple IIc Parallel	$49.99
Kaypro 2000 Parallel	$49.99
C64/128	$59.99

Orange Micro

Grappler CD (C64)	$89.99
Grappler Plus (IIe, IIc)	$89.99
Grappler C (IIc)	$89.99
Grappler 16K (IIe, II+)	$139.00

PRACTICAL PERIPHERALS

Graphcard	$69.99
Serial Card	$99.99
Microbuffer II+ 64K	$169.00

QUADRAM

Microfazer	from $139.00
Efazer (Epson)	from $79.99

PRINTERS

Canon

A40,A50,A55	CALL
LBP-8A1 Laser	CALL

CITIZEN

MSP-10 (80 col.)	$279.00
MSP-15 (132 col.)	$389.00
MSP-20 (80 col.)	$349.00
MSP-25 (132 col.)	$509.00

C.ITOH

Prowriter 7500	$169.00
Prowriter 1550P	$349.00
Starwriter 10-30	$399.00
3500 Tri Printar	$1499.00

corona

Lazer LP-300	$2799.00

DIABLO

620 Daisywheel	$299.00
D25 Daisywheel	$549.00
635 Daisywheel	$1099.00
D80IF Daisywheel	CALL

daisywriter

2000	$699.00

EPSON

Homewriter 10, LX-80	CALL
FX-85, FX-286, RX-100	CALL
DX-10, DX-20, DX-35	CALL
SQ-2000, Hi-80, HS-80, AP-80	CALL
LQ-600, LQ-1000	CALL

JUKI

6000 Letter Quality	CALL
6100 Letter Quality	CALL
6200 Letter Quality	CALL
6300 Letter Quality	CALL
6500 Letter Quality	CALL
5510 Dot Matrix Color	CALL

LEOENO

808 Dot Matrix 100 cps	$179.00
1080 Dot Matrix 100 cps	$259.00
1380 Dot Matrix 130 cps	$289.00
1385 Dot Matrix 165 cps	$339.00

NEC

3000 Series	$779.00
8000 Series	$1099.00
ELF 360	$399.00
Pinwriter 580	$999.00

OKIDATA

182, 163, 192, 193, 2410, 64	CALL
Qkimate 10 (Specify C64/Atari)	$189.00
Qkimate 20 (IBM)	CALL

Panasonic

KX1080	NEW
KX1091	$259.00
KX1092	$389.00
KX1592	$489.00
KX1595	$659.00

QUADRAM

Quadjet	$399.00
Quad Laser	CALL

SILVER-REED

500 Letter Quality	$219.00
550 Letter Quality	$419.00
600 Letter Quality	$699.00

star

SG-10C (C64 Interface)	CALL
SB/SD/SG/SR Series	CALL
Powertype Letter Quality	CALL

Texas Instruments

TI850	$529.00
TI855	$639.00
TI865	$799.00

TOSHIBA

1340 (80 column)	$369.00
P341 (132 column)	$799.00
P351 (132 column)	$1049.00

SOFTWARE FOR IBM

ANBA SOFTWARE

Paradox	$499.00

ASNTON-TATE

Framework II	$389.00
dBase III Plus	$389.00

BATTERIES INCLUDED

Isgur Portfolio	$159.00

BORLAND

Lightening	$59.99
Sidekick (unprotected)	$57.99
Reflex	$59.99
Travelling Sidekick	$44.99

CENTRAL POINT

Copy II PC-Backup	$29.99
PC Option Board	$84.99

DECISION RESOURCES

Chartmaster	$229.00
Signmaster	$159.00
Diagram Master	$209.00

FIFTH GENERATION

Fast Back	99.99

FUNK SOFTWARE

Sideways	$44.99

NARVARD SOFTWARE INC.

Total Project Manager	$289.00
Presentation Graphics	$239.00

INFOCOM

Cornerstone	CALL

LIFETREE

Volkswriter III	$159.00

LIVING VIDEOTEXT

Think Tank	$109.00
Ready	$64.99

LOTUS

Symphony	CALL
1-2-3 Version 2	CALL

MECA SOFTWARE

Managing Your Money 2.0	$99.99
Manage Your Market	$89.99

MICROPRO

Easy	$94.99
WordStar 2000	$239.00
WordStar 2000+	$289.00
WordStar Professional	$189.00

MICRORIM SOFTWARE

R:Base 4000	$249.00
R:Base 5000	$389.00
Clout 2.0	$129.00

MICROSOFT

Flight Simulator	$34.99
MultiPlan	$129.00
Word	$249.00
Mouse	$139.00

MICROTUF

Crosstalk XVI	$89.99
Crosstalk Mark IV	$149.00
Remote	$89.99

MULTIMATE

Multi Mate Word Proc	$219.00
Advantage	$289.00
On File	$89.99
Just Write	$89.99

NOUNEMON

Intuit	$69.99

NORTON

Norton Utilities 3.1	$57.99

ONE STEP

Golf's Best	$34.99

PFS: IBM

Proof	$59.99
File/Graph (ea)	$84.99
Report	$74.99
Write/Proof Combo	$84.99

PROFESBIONAL SOFTWARE

Write-N-Spell	$89.99

THE BOFTWARE OROUP

Enable	$329.00

SATELLITE BYBTEMS

Word Perfect 4.1	$219.00

SORCIM/IUS

Accounting AP/AR/GL/INV/QE (ea.)	$299.00
SuperCalc III	$199.00
EasyWriter II System	$239.00
Super Project	$199.00

BUBLOOIC

Jet	$37.99

IBM

IBM PC SYSTEMS

Configured to your specifications.

Call for Best Price!

IBM-PC, IBM-XT, IBM-AT

AT&T

Safari (7300)	CALL
6300	CALL

corona

PPC400 Dual Portable	$1289.00
PPCXT 10 meg Portable	$1989.00
PC40022 Dual Desktop	$1389.00
PC400-HD2 10 meg	$1989.00

ITT

ITT X-TRA 256K, 2 Drive System	CALL
256K,10 meg Hard Drive System	CALL
XP5, 20 meg	CALL

KAYPRO

KP-2000 Portable	CALL
Kaypro PC	CALL

SANYO

MBC 550-2, MBC 555-2, MBC 675 Portable, MBC775, MBC 880 DesktopCALL

SPERRY

Sperry-AT	as low as $1749.00
Sperry-IT	as low as $2699.00
Call for Specific Configuration!	
All Models	CALL

ZENITH

PC-138 Series, PC-148 Series, PC-158 Series, PC-160 Series, PC-171 Series, AT-200 Series	CALL

MULTIFUNCTION CARDS

AST

RamVantage	$349.00
Rampage-PC	$379.00
Rampage-AT	CALL
Six Pack Plus	$229.00
I/Q Plus II	$139.00
Advantage-AT	$399.00
Preview Mono	$299.00
PC Net Cards	$379.00
5251/11 On-line	$669.00
5251/12 Remote	$579.00

dca

IRMA 3270	$879.00
IRMA Print	$999.00
IRMA Smart Alec	$779.00

EVEREX

Edge Card	$259.00
Graphics Edge	$219.00
Magic Card II	$159.00
Magic Card I	$99.99

HERCULES

Graphics	$299.00
Color	$159.00

IDEAssociates

IDEA 5251	$549.00

INTEL

PCNC8087 5MHz	
PCNC80287-2 8 MHz	CALL
PCNC80287 6 MHz	FOR
1010 PC-Above Board	YOUR
1110 PS-Above Board	PC
2010 AT-Above Board	

MYLEX

The Chairman	$439.00

PARADISE

Color/Mono Card	$149.00
Moduler Graphics Card	CALL
Multi Display Card	$199.00
Five Pack C, S	$99.99
High Res Mono	CALL

PERSYST

Bob Board	$359.00

QUAORAM

Quadport-AT	$119.00
Liberty-AT (128K)	$349.00
The Gold Quadboard	$449.00
The Silver Quadboard	$239.00
Expanded Quadboard	$199.00
Liberty	$309.00
QuadSprint	$499.00
QuadLink	$399.00
QuedColor	$199.00
Quadboerd AT	$399.00
8600 E G A card	$399.00

TECMAR

Captain - 64	$199.00
Graphics Master	$469.00

Model Name	Manufacturer/ Distributor	Compatibility	Print Technology	Speed	Pitch	Buffer	Feed Type	Warranty	Suggested Retail Price	Comments
BJ-80	Canon USA. Inc.	Parallel std	Bubble jet	110–220 cps	10 cpi std	2–4.3K	Pin std	1 year	599	Cross between ink jet and thermal transfer
PJ-1080A	Canon USA, Inc	Parallel std	Ink jet (color)	37 cps	12 cpi std	1 line	Friction std	1 year	699	
H-80	Centronics	Parallel std, serial opt	Dot matrix	27–160 cps	5–20 cpi	2K opt	Cut sheet and fan fold std	1 year	699	
Citizen 120D	Citizen America Corp	Parallel std, serial opt	Dot matrix	25-120 cps	5-20 cpi	4K	Tractor std	1 year	249	
MSP-10	Citizen America Corp	Parallel std, serial opt	Dot matrix	40–160 cps	10–12 cpi	1K	Friction and tractor std, cut sheet feeder opt	18 months	399	Emulates IBM graphics
MSP-15	Citizen America Corp	Parallel std, serial opt	Dot matrix	40–160 cps	10 cpi	1K	Tractor and friction std, cut sheet feeder opt	18 months	549	
MSP-20	Citizen America Corp	Parallel std, serial opt	Dot matrix	50–100 cps	10–12 cpi	8K	Friction and tractor std, cut sheet feeder opt	18 months	499	Can create own graphics
Premiere 35	Citizen America Corp	Parallel std, serial opt	Daisy wheel	35 cps	10-15 cpi	8K	Tractor std	1 year	599	Diablo print wheel
MPS 1000	Commodore Business Machines	Parallel and serial std	Dot matrix	20–100 cps	12–17 cpi	1K	Friction and tractor std	90 days	299.95	
Printelex	Computer Peripherals	Parallel and serial std	Dot matrix	160 cps	12 cpi	1 line	Friction std	90 days	157.50	40 column comes with power pack. 1 1/2 pounds
Dataproducts 8010	Dataproducts Corp	Parallel and serial std	Dot matrix	30–180 cps	10–17 cpi	2K (8K opt)	Friction and tractor std	1 year	499	Block or dot addressable graphics
Dataproducts 8012	Dataproducts Corp	Parallel std	Dot matrix	20–180 cps	10–17 cpi	2K std (8K opt)	Friction and tractor std	1 year	499	Block or dot addressable graphics
FORTIS DX-15XL	Dynax	Parallel std, serial opt	Daisy wheel	20 cps	10–15 cpi	5K	Friction std, cut sheet feeder and tractor opt	90 days	599	
FORTIS DX-25	Dynax	Parallel and serial std	Daisy wheel	25 cps	10–15 cpi	7K	Friction and tractor std, cut sheet feeder opt	90 days	645.95	
AP-80	Epson America, Inc	Apple exclusive	Dot matrix	15–75 cps	9-17 cpi	1K	Friction and tractor std, auto sheet feeder opt	1 year	379	
DX-10	Epson America, Inc.	Parallel std	Daisy wheel	10 cps	10–12 cpi	None	Friction, tractor and cut sheet feeder opt	1 year	299	
DX-20	Epson America, Inc	Diablo all purpose interface std	Daisy wheel	20 cps	10-15 cpi	1K (7K opt)	Friction std, tractor and cut sheet feeder opt	1 year	459	110 column
FX-85	Epson America, Inc.	Parallel std	Dot matrix	32–160 cps	N/A	8K	Friction and tractor std, cut sheet feeder opt	1 year	549	
FX-286	Epson America, Inc	Parallel std	Dot matrix	40-200 cps	5-20 cpi	8K	Friction and tractor std, cut sheet feeder opt	1 year	799	
Homewriter 10	Epson America, Inc.	PIC's (Printer Interface Cartridges) are available for most home computers)	Dot matrix	16–100 cps	5-20	1K	Friction std. tractor and cut sheet opt	1 year	249	Selectype available
HS-80	Epson America, Inc	Parallel std	Ink jet	32–160 cps	5-20 cpi	1K	Friction, auto cut sheet feeder opt	1 year	499	
JX 80	Epson America, Inc	Parallel std	Dot matrix	160 cps	10–12 cpi	2K	Friction and tractor std	1 year	399	
LQ800	Epson America, Inc.	Parallel and serial std	Dot matrix	60-180 cps	10-15 cpi and proportional	7K	Friction std	1 year	799	24-pin printhead
LX-80	Epson America, Inc	Parallel std	Dot matrix	16–100 cps	4 3-20 cpi	1K (32K opt)	Friction std. tractor and cut sheet feeder opt	1 year	329	Over 160 typestyles available through Selectype
LX-90	Epson America, Inc	PIC	Dot matrix	16–100 cps	5-20 cpi	1K	Friction and tractor std	1 year	329	
CP-80 Type 1	Everett/Charles Marketing Service, Inc.	Parallel std, serial opt	Dot matrix	80 cps	N/A	None	Friction and tractor std	90 days	175	Bit image graphics; Superscript
Facit 4509	Facit, Inc	Parallel std	Dot matrix	70–120 cps	10–17 cpi and proportional	None	Tractor std	90 days	425	IBM compatible graphics
Facit 4510	Facit. Inc	Parallel and serial std	Dot matrix	70–120 cps	10–17 cpi and proportional	2K	Friction and tractor std	90 days	495	Block and pin addressable graphics
Facit 4511	Facit. Inc	Parallel and serial std	Dot matrix	40–160 cps	10–17 cpi and proportional	2K	Friction and tractor std	90 days	595	Wide carriage version $795
Facit 4513	Facit. Inc	Parallel and serial std	Dot matrix	40–160 cps	10–17 cpi and proportional	2K	Friction and tractor std	90 days	695	
Facit D2000	Facit. Inc	Parallel or serial std	Daisy wheel	24–30 cps	10–15 cpi and proportional	2K	Friction std. tractor opt	90 days	695	
DX2100	Fujitsu America, Inc	Parallel std, serial opt	Dot matrix	220 cps	10–17 cpi	2K (18K opt)	Friction and tractor std, cut sheet feeder opt	1 year	495 (Color 645)	Dot addressable graphics
DX2200	Fujitsu America. Inc	Parallel std. serial opt	Dot matrix	220 cps	10–17 cpi	7K std (16K opt)	Friction and tractor std, cut sheet feeder opt	1 year	645	

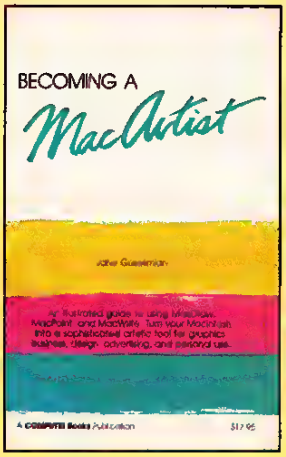
Model Name	Manufacturer/ Distributor	Compatibility	Print Technology	Speed	Pitch	Buffer	Feed Type	Warranty	Suggested Retail Price	Comments
GE 8100	General Electric	Parallel std; Atari, Commodore and IBM PCjr interfaces available	Thermal transfer (non-impact dot matrix)	25–50 cps	24 cpi	2K	Friction std	2 years	259 95	Block graphics, special graphics characters
MP-1300AI	Hatton Seiko	Parallel and serial std	Dot matrix	300 cps	10-20 cpi	10K (7K when downloadable characters)	Friction and tractor std	2 years	799	Optional color printer kit for under $200
SP-1000	Hatton Seiko	Parallel or serial std	Dot matrix	100 cps	10–17 cpi and proportional	Varies with model	Friction and tractor std	2 years	299	Commodore/VIC version $270
Thinkjet (HP2225)	Hewlett-Packard	Parallel, HP-1B, HP-1L and serial available	Thermal ink jet	150 cps	5–18 cpi	1K	Friction and pin std	1 year	495	Best results using special paper, portable
Juki 5510	Juki Office Machines Corp	Parallel std	Dot matrix	20–180 cps	10–17 cpi	3K (15K opt)	Friction and tractor std	1 year	499	
Juki 6000	Juki Office Machines Corp	Parallel std; serial opt	Daisy wheel	10 cps	10–15 cpi	2K-serial	Friction std	1 year	295	
Juki 6100	Juki Office Machines Corp	Parallel std, serial opt	Daisy wheel	17 cps	10–15 cpi and proportional	2K	Friction std, tractor and cut sheet feeder opt	N/A	599	
Juki 6200	Juki Office Machines Corp	Parallel std, serial opt	Daisy wheel	30 cps	10–15 cpi and proportional	3K std (15K opt)	Friction std, tractor and cut sheet feeder opt	1 year	745	
Westrex One	Litton Westrex	Parallel or serial std	Dot matrix	42–140 cps	10–15 cpi	2K	Friction and pin std	1 year	199	
MT85	Mannesmann Tally	Parallel, serial or Apple std	Dot matrix	45–180 cps	10–17 cpi	3K	Friction and tractor std	1 year	499	Dot addressable graphics
MT 86	Mannesmann Tally	Parallel, serial, or Apple std	Dot matrix	45–180 cps	10–17 cpi	N/A	Friction and tractor std	1 year	599	
PC-PR105A	NEC Home Electronics, Inc	Parallel std	Dot matrix	46–92 cps	10 cpi	4K	Friction and tractor std	90 days labor, 1 year parts	399	
E L F 350	NEC Information Systems, Inc	Parallel and serial std	Daisy wheel	19 cps	10–15 cpi	2K	Cut sheet guide std, cut sheet feeder and tractor opt	1 year	545	
E L F 360	NEC Information Systems, Inc	Parallel and serial std	Daisy wheel	19 cps	10–15 cpi	2K	Cut sheet guide std, cut sheet feeder and tractor opt	1 year	545	
P2	NEC Information Systems, Inc	None std; IBM serial, serial, IBM parallel, and parallel opt	Dot matrix	35–180 cps	10–17 cpi	3 5K–5K	Tractor std; cut sheet feeder opt	1 year	699	
Microline 182	Okidata	Parallel std; serial opt	Dot matrix	30-120 cps	5–17 cpi	1 line	Friction and pin std	1 year	299	Block and bit image graphics
Microline 182 TTY	Okidata	Parallel std, serial opt	Dot matrix	30-120 cps	5-17 cpi	1 line	Friction and pin std	1 year	349	Designed for communications applications, 4 levels of intelligence
Microline 183	Okidata	Parallel std,serial opt	Dot matrix	30-120 cps	5–17 cpi	1 line	Friction and tractor std	1 year	549	Wide carriage version of 183, block and bit image graphics
Microline 192	Okidata	Parallel std; serial opt	Dot matrix	33–160 cps	5–17 cpi	8K	Friction and pin std	1 year	499	Block and bit image graphics
Microline 193	Okidata	Parallel and serial std	Dot matrix	33–160 cps	5–17 cpi	8K	Friction and tractor std	1 year	699	Wide carriage version of 192, block and bit image graphics
Microline 292	Okidata	Parallel or serial std	Dot matrix	100-200 cps	10-17 cpi	8K	Pin std, tractor and cut sheet feeder opt	1 year	699	
Okidata 120	Okidata	Commodore serial	Dot matrix	30-120 cps	5–17 cpi	1 line	Friction and pin std	1 year	269	All points addressable graphics, Commodore Special Graphics
Okimate 10	Okidata	Commodore and Atari	Dot matrix	60 cps	5–17 cpi	1 line	Friction and pin std	90 days	208	All points addressable graphics; Commodore Special Graphics
Okimate 20	Okidata	IBM, Apple and Amiga	Dot matrix	40–80 cps	5–17 cpi	8K	Friction and tractor std	90 days	268	High resolution, all points addressable, bit image graphics
KX-P1080	Panasonic Co	Parallel std; serial opt	Dot matrix	20–100 cps	10–17 cpi	1K	Friction and tractor std	2 years	319	Bit image graphics, can emulate Image Writer, Epson RX-80 compatible, color ribbons available
KX-P1091	Panasonic Co	Parallel std; serial opt	Dot matrix	29–120 cps	10–17 cpi	1K (4K opt)	Friction and tractor std	2 years	399	Same as above
KX-P1092	Panasonic Co	Parallel std; serial opt	Dot matrix	22–180 cps	10–12 cpi and proportional	7K	Friction and push tractor std	2 years	499	Bit image graphics, Epson FX-80 compatible, color ribbons available
KX-P1592	Panasonic Co	Parallel std; serial opt	Dot matrix	38–180 cps	10–17 cpi and proportional	7K text std (32K opt)	Push tractor or friction std	2 years	699	Colored ribbon available
KX-P3131	Panasonic Co	Parallel std; serial opt	Daisy wheel	17 cps	N/A	6K (32K opt)	Friction std, tractor and auto cut sheet feeder opt	2 years	419	Diablo 630 code compatible, color ribbons available
KX-P3151	Panasonic Co	Parallel std; serial opt	Daisy wheel	22 cps	10–12 cpi	7K std (54 5K opt)	Friction std, tractor and cut sheet feeder opt	2 years	659	

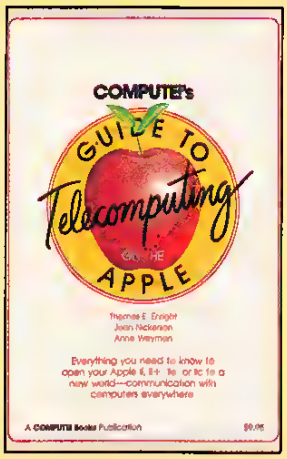
Model Name	Manufacturer/ Distributor	Compatibility	Print Technology	Speed	Pitch	Buffer	Feed Type	Warranty	Suggested Retail Price	Comments
RP2200Q	Ricoh Corporation	Parallel and serial std	Daisy wheel	20-22 cps	10-15 cpi	One line	Friction std; auto sheet feeder and tractor opt	90 days	699	
Letter Master	Royal Consumer Business Products	Parallel std; serial opt	Daisy wheel	10 cps	10-12 cpi	84 characters	Friction std	90 days labor, 1 year parts	349.95	
OfficeMaster 2000	Royal Consumer Business Products	Parallel and serial std	Daisy wheel	23 cps	10-15 cpi and proportional	1.5K	Friction std; tractor and sheet feeder opt	90 days labor, 1 year parts	599.95	
SP 1200 PLUS	Sakata	Parallel std	Dot matrix	120 cps	5-16.5	N/A	Friction and pin std	90 days	399	
SP 1500	Sakata	Parallel std	Dot matrix	180 cps	5-16.5	N/A	N/A	90 days	489	
SP 5500	Sakata	Parallel std; serial opt	Dot matrix	36-180 cps	5-18 cpi	3K	Friction and tractor std	90 days	699	Wide carriage
EXP 500	Silver-Reed, Inc.	Parallel or serial std	Daisy wheel	14 cps	10-15 cpi	None	Friction std; tractor and cut sheet feeder opt	90 days	449	
EXP 550	Silver-Reed, Inc.	Parallel or serial std	Daisy wheel	19 cps	10-15 cpi and proportional	None	Friction std; cut sheet feeder and tractor opt	90 days	649	
EXP 600	Silver-Reed, Inc.	Parallel or serial std	Daisy wheel	25 cps	10-15 cpi and proportional	3K std (19K and 40K opt)	Friction std; tractor and sheet feed opt	90 days labor, 1 year parts	N/A	June release
Silver-Reed 400	Silver-Reed, Inc.	Parallel or serial std	Daisy wheel	10 cps	10-15 cpi	None	Friction std; tractor opt	90 days	249	
NX-10	Star Micronics	Parallel std	Dot matrix	30-120 cps	5-10 cpi	5K	Friction and tractor std	1 year	349	Emulates IBM graphics printer
Powertype	Star Micronics	Parallel std; serial opt	Daisy wheel	18 cps	10-15 cpi and proportional	1 line	Friction std; tractor opt	180 days	499	
SD-10	Star Micronics	Parallel std; serial opt	Dot matrix	160 cps	N/A	2K	Friction and tractor std	1 year	449	Ultra-high resolution bit image graphics
SD-15	Star Micronics	Parallel std; serial opt	Dot matrix	160 cps	N/A	16K	Friction and tractor std	1 year	599	
SR-10	Star Micronics	Parallel std; serial opt	Dot matrix	200 cps	N/A	2K	Friction, tractor, automatic single sheet feed std	1 year	649	
SR-15	Star Micronics	Parallel std; serial opt	Dot matrix	200 cps	N/A	2K	Friction, tractor, single sheet feeder std	1 year	799	15" carriage
STX-80	Star Micronics	Parallel std	Thermal transfer dot matrix	60 cps	5-11 cpi	1 line	Friction std	1 year	199	Dot addressable bit image graphics
Compumate 2100	Swintec Corporation	Parallel and serial std	Daisy wheel	16 cps	10-15 cpi	1 line	Friction std	90 days	599	
M-20P	TAB Products	Parallel std	Dot matrix	120 cps	10-15 cpi	1K	Friction and pin tractor opt	90 days	299	Dot addressable graphics
M-21P	TAB Products	Serial std	Dot matrix	120 cps	10-15 cpi	1K	Friction and pin tractor opt	90 days	378	Dot addressable graphics
M-22P	TAB Products	Parallel std	Dot matrix	160 cps	10-15 cpi	1K	Friction and pin std; tractor and cut sheet feeder opt	90 days	499	Dot addressable graphics
M-23S	TAB Products	Serial std	Dot matrix	160 cps	10-15 cpi	1K	Pin and friction std; cutsheet feeder and tractor opt	90 days	549	
M-24P	TAB Products	Parallel std	Dot matrix	160 cps	10-15 cpi	1K	Tractor and friction std; cut sheet feeder opt	90 days	699	
CGP 220	Tandy Corp	Parallel and serial std	Ink jet	37 cps	12 cpi	1 line	Friction std	90 days	599	
DMP 105	Tandy Corp	Parallel and serial std	Dot matrix	80 cps	10-17 cpi and proportional	N/A	Friction and tractor std	90 days	199.95	Bit image graphics
DMP 130	Tandy Corp	Parallel and serial std	Dot matrix	100 cps	10-16 cpi	N/A	Friction and tractor std	90 days	349.95	Bit image graphics
DWP 220	Tandy Corp	Parallel and serial std	Daisy wheel	20 cps	10 12 and proportional	N/A	Friction std; tractor opt	90 days	599	
TRP 100	Tandy Corp	Parallel and serial std	Dot matrix	50 cps	N/A	N/A	Friction and sheet feed std	90 days	199.95	Uses thermal paper
IMP-24	Weigh-Tronix, Inc.	Parallel or serial std	Dot matrix	16.8 cps	N/A	1 line	Friction std	90 days	135 (24 cpl) 150 (32 or 40 cpl)	Dot addressable graphics
Companion 10!	Xerox/Diablo	Parallel std	Dot matrix	80 cps	10-15 cpi	1K	Friction and tractor std	90 days	399	Bit map graphics
Companion 12CQ1	Xerox/Diablo	Parallel std	Dot matrix	60-150 cps	10-15 cpi	2K	Friction and tractor std	90 days	599	

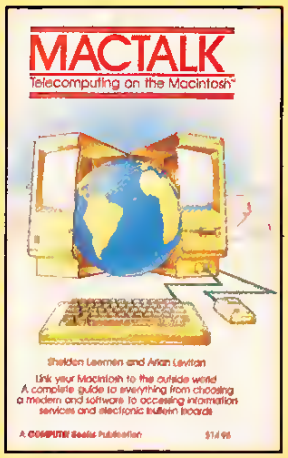
APPLE TITLES

BACK BY POPULAR DEMAND!

These popular titles for the Apple-ii series and Macintosh computers contain expert programming advice, in-depth tutorials, valuable buying guides, challenging business, home, and educational applications, and exciting games. They're all written in COMPUTE!'s easy-to-understand style to provide you with hours of creative entertainment and information.







Becoming a MacArtist

Vahé Guzelimian, 312 pages
An illustrated guide to Macintosh graphics with which you can explore the full power and flexibility of *MacDraw*, *MacPaint*, and *MacWrite*.

$17.95 ISBN 0-942386-80-9

COMPUTE!'s Apple Games for Kids

Clark and Kathy H. Kidd, 305 pages
An instant library of educational software including simple tests, music and graphics programs, a typing tutor, menu planner, and game-writing utilities for the whole family.

$12.95 ISBN 0-942386-91-4

COMPUTE!'s Guide to Telecomputing on the Apple

Thomas E. Enright, Joan Nickerson, and Anne Wayman, 173 pages
An informative and easy-to-understand guide to telecomputing on the Apple, from selecting hardware and software to accessing large databases.

$9.95 ISBN 0-942386-98-1

COMPUTE!'s Kids and the Apple

Edward H. Carlson, 255 pages
An entertaining and easy-to-use book for kids ages 10–14 interested in learning BASIC programming on their Apple computers.

$12.95 ISBN 0-942386-76-0

MacTalk: Telecomputing on the Macintosh

Sheldon Leemon and Arlan Levitan, 263 pages
A complete guide to telecomputing on the Macintosh, from choosing a modem and software to accessing information services and electronic bulletin boards.

$14.95 ISBN 0-942386-85-X

COMPUTE!'s First Book of Apple

Edited, 228 pages
A collection of 35 exciting games, educational programs, home applications, and graphics routines for owners of Apple II, IIe, and IIc computers.

$12.95 ISBN 0-942386-69-8

Easy BASIC Programs for the Apple

Brian Flynn, 390 pages
A wide-ranging selection of short BASIC programs—for home, school, and office—for Apple II-series computers.

$14.95 ISBN 0-942386-88-4

Pick your favorite titles and order today!
Call toll-free 800-346-6767 (in NY 212-887-8525).
Or mail your payment with $2.00 shipping and handling per book to COMPUTE! Books, P.O. Box 5038, F.D.R. Station, New York, NY 10150.

Please allow 4–6 weeks for delivery.

COMPUTE! Publications, Inc.
Part of ABC Consumer Magazines Inc
One of the ABC Publishing Companies
825 7th Avenue 6th Floor New York NY 10019
Publishers of COMPUTE! COMPUTE!'s Gazette COMPUTE!'s Gazette Disk COMPUTE! Books and COMPUTE!'s Apple Applications

COMPUTE! books are available in the U.K., Europe, the Middle East, and Africa from Holt Saunders, Ltd., 1 St Anne's Road, Eastbourne, East Sussex BN21 3UN, England, and in Canada from McGraw-Hill, Ryerson Ltd., 330 Progress Ave., Scarborough, Ontario, Canada M1P 2Z5.

For more information on any of the printers listed here, please contact:

Alphacom
2323 S. Bascom Avenue
Campbell, CA 95008

Apple Computer
20525 Mariani Avenue
Cupertino, CA 95014

Aprotek
1071-A Avenida Acaso
Camarillo, CA 93010

Axiom
1014 Griswold Avenue
San Fernando, CA 91340

Axonix
417 Wakara Way
5alt Lake City, UT 84108

Blue Chip Electronics
2 W. Alameda Drive
Tempe, AZ 85282

Brother International
8 Corporate Place
Piscataway, NJ 08854

C. Itoh Digital Products
19750 S. Vermont Avenue
5uite 220
Torrance, CA 90502

CAL-ABCO
6041 Variel Avenue
Woodland Hill, CA 91367

Canon USA
5ystem Division
One Canon Plaza
Lake 5uccess, NY 11042

Centronics Data Computer
1 Wall Street
Hudson, NH 03051

Citizen America
2425 Colorado Avenue #300
Santa Monica, CA 90404

Commodore Business Machines
1200 Wilson Drive
West Chester, PA 19380

Computer Peripherals
6400 Canoga Avenue
Suite 305
Woodland Hills, CA 91367

Dataproducts
6200 Canoga Avenue
Woodland Hills, CA 91365

Dynax
6070 Rickenbacker Road
Commerce, CA 90040

Epson America
3415 Kashiwa 5treet
Torrance, CA 90505

Everett/Charles Marketing 5ervices
6101 Cherry Avenue
Fontana, CA 92335

Facit
9 Executive Park Drive
P.O. Box 334
Merrimack, NH 03054

Fujitsu America
3055 Orchard Road
San Jose, CA 95134

General Electric
Electronics Park
5yracuse, NY 13221

Hattori 5eiko Computer Peripherals
Division of Hattori Corporation of America
1111 Macarthur Boulevard
Mahwah, NJ 07430

Hewlett-Packard
1100 Wolfe Road
Cupertino, CA 95014

Juki Office Machine
20437 S. Western Avenue
Torrance, CA 90501

Litton Westrex
Printer Division
100 Riggenback Road
Fall River, MA 02720

Mannesmann Tally
8301 S. 180th
Kent, WA 98032

NEC Home Electronics (USA)
Personal Computer Division
Elk Grove Village, IL 60007

NEC Information 5ystems
1414 Massachusetts Avenue
Boxborough, MA 01719

Okidata
532 Fellowship Road
Mt. Laurel, NJ 08054

Panasonic
Computer Products Division
One Panasonic Way
5ecaucus, NJ 07094

Ricoh America
5 Dedrick Place
West Caldwell, NJ 07006

Royal Consumer Business Products
500 Day Hill Road
Windsor, CT 06095

5akata U5A
651 Bonnie Lane
Elk Grove Village, IL 60007

Silver-Reed America
19600 5. Vermont Avenue
Torrance, CA 90502

5tar Micronics Peripheral Division
200 Park Avenue
5uite 2309
Pan Am Bldg.
New York, NY 10166

Swintec
23 Poplar 5treet
P.O. Box 421
East Rutherford, NJ 07073

TAB Products
1400 Page Mill Road
Palo Alto, CA 94304

Tandy /Radio Shack
1800 One Tandy Center
Fort Worth, TX 76102

Weigh-Tronix
1000 N. Armstrong
Fairmont, MN 56031

Xerox/Diablo
901 Page Avenue
P.O. Box 5030
Fremont, CA 94537

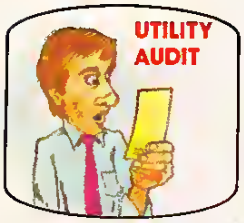
Get the jump on the weather-man by accurately forecasting the local weather yourself!

A scientifically proven way to develop an awesome memory.

You are trapped in a five-story, 125-room structure made entirely of ice. Find the exit before you freeze!

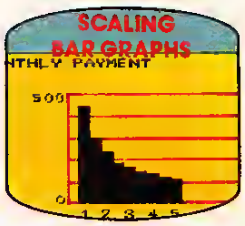
Take control of your personal finances in less than one hour a month.

The beautiful princess is held captive by deadly dragons. Only a knight in shining armor can save her now!

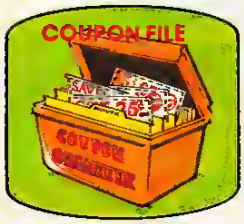


Cut your energy costs by monitoring your phone, electric and gas bills.

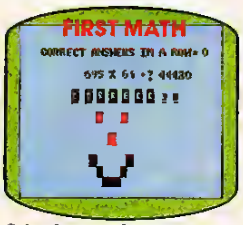
Computerize car maintenance to improve auto performance, economy and resale value.



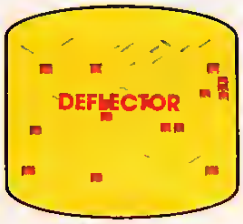
Create multi-colored bar graphs with a surprisingly small amount of memory.



A time-saving organizer for coupons, receipts and more.



School-age and pre-school children are rewarded for right answers, corrected on their wrong ones.



A real brainflexer. Deflect random balls into targets on a constantly changing playfield.

A fun way to dramatically increase typing speed and accuracy.

Get up to 20 new programs and games every month in COMPUTE!

Every month, COMPUTE! readers enjoy up to 20 brand-new, ready-to-run computer programs, even arcade-quality games.

And when you subscribe to COMPUTE!, you'll get them all for less than 15 cents each!

You'll find programs to help you conserve time, energy and money. Programs like Cash Flow Manager, Retirement Planner, Coupon Filer, Dynamic Bookkeeping.

You'll enjoy games like Air Defense, Boggler, Slalom, and High Speed Mazer.

Your children will find learning fast and fun with First Math, Guess That Animal, and Mystery Spell.

Looking for a challenge? You can write your own games. Customize BASIC programs. Even make beautiful computer music and pictures.

It's all in COMPUTE!. All ready to type in and run on your Atari, Apple, Commodore, TI-99/4A, IBM PC, or PCjr computers.

What's more, you get information-packed articles, product reviews, ideas and advice that add power and excitement to all your home computing.

And when it's time to shop for peripherals or hardware, check COMPUTE! first. Our product evaluations can save you money and costly mistakes. We'll even help you decide what to buy: Dot-matrix or daisy-wheel printer? Tape storage or disk drive? What about modems? Memory expansion kits? What's new in joysticks, paddles, and track balls?

Order now! Mail the postpaid card attached to this ad and start receiving every issue of COMPUTE!.

For Faster Service Call Toll-Free 1-800-247-5470 (in Iowa 1-800-532-1272)

COMPUTE! P.O. Box 10954, Des Moines, IA 50340

MIAMI ICE

Jeff Kulczycki

Here's an action game that challenges both your driving skills and powers of concentration. Originally written for the Commodore 128 with a disk drive, "Miami Ice" has been translated to work on the Commodore 64, Atari 400/800/XL/XE (with at least 32K RAM), and Apple II-series computers. A joystick is required.

Ah, Miami—sun city of the South. A sparkling metropolis blessed with a tropical climate, palm trees, beaches, revived art deco architecture, stylish pastels, and classy elegance. Almost paradise.

You wake up on another bright, sunny Miami morning, sip a glass of freshly squeezed orange juice, don your white linen suit and sunglasses, and stroll outside—then get the shock of your life.

What's going on here? Overnight, a freak shift in the jet streams has piped a blistering cold front down from Ohio. The weatherman had predicted a brief shower last evening, but that's not what happened. Instead, the Florida peninsula was blasted by the worst ice storm in 400 years. The Everglades are frozen solid. The pink flamingos are blue. And the streets of Miami are coated with a shimmering layer of slippery ice.

As you start your car—the pampered engine coughs and sputters in the bitter cold—you wonder what it's going to be like driving to work. A Miami native, you've never driven on ice before. In fact, you've never even *seen* this much ice since your boss's retirement party last year, when the caterers made that life-size ice sculpture of Ponce de Leon. You've heard the horror stories told by tourists about winter driving conditions up North, but never thought it could happen to you—not here, in Miami.

The minute you pull out onto the street, your worst fears come true. When you step on the gas pedal, the wheels spin and the car accelerates sluggishly. When you turn the steering wheel, the car slides all over the road. And when you step on the brakes—well, forget it.

You realize, desperately, that you've got to make it to the parking garage across town without smashing your car to smithereens. It won't be easy. But at least there's one thing in your favor—you've got the whole road to yourself.

The Source℠ Is Friendly.

Many online information services claim to be "user friendly." But only one really lives up to that promise.

The Source.

You see, The Source is specifically designed to save you time online. With new, shorter menus. Simpler commands. And a user's manual so well-written and easy to understand, it's setting an industry standard.

The Source is also the only service that gives you introductory lessons and assistance, free of on-line charges. So you get up-to-speed on our dime, not yours.

You get to the information you need in record time, without frustration. Everything from the hour's headlines to travel reservations. From special interest groups to online stock trading. So you don't waste your valuable time. Or money.

Call 1-800-336-3366, send the coupon, or visit your nearest computer dealer.

And make friends with America's friendliest online information network.

INFORMATION NETWORK

The Source℠

The most powerful resource any personal computer can have.

I'd like to make friends with The Source. Please send more information to:

Name

7826J24

Address

City, State Zip

Mail to: Source Telecomputing Corporation, 1616 Anderson Road, McLean, VA 22102

Everyone else, it seems, had the good sense to stay home.

Out Of Control

Despite minor variations, all four versions of "Miami Ice" work basically the same. Using a joystick, you have to drive your car over ice-covered streets to reach the safety of a garage. The joystick button is the gas pedal, and pushing the stick right or left steers the car in the corresponding direction.

But here's the twist—the car doesn't respond instantly to your commands. It tends to slide in the same direction even after you've steered it toward another direction. Then, when you try to recover, you often overcorrect and start sliding in yet another new direction. It's an inertial nightmare—much like real winter driving.

When you hit a guardrail or some other obstruction, your car cracks up. You get three cars per game. If you reach the safety of the garage, the game isn't over. Instead, you advance to another screen whose streets are even harder to navigate.

The number of points you score depends on how soon you reach the garage. As an incentive to recklessness, a timer starts counting down when you begin each new screen. If you reach the garage, you score the number of points left on the timer. If the timer runs out, you can still reach the garage, but you won't get any points. However, you will advance to the next screen.

Be sure to read the special instructions for each version before typing in the program and playing the game.

Commodore 128

The 128 version of Miami Ice (Program 1) is written completely in BASIC using BASIC 7.0's excellent sprite commands. It runs as fast and as smoothly as the other versions, which all employ machine language.

Plug a joystick into port 2 and leave a disk in the drive. After each game, if your score ranks you among the top players, the program lets you enter your initials and then saves the high score data to disk.

To complete each level, you merely have to steer your car into the parking garage from any angle. There are a total of four screens, and each screen displays the timer value in the upper-left corner and your current score immediately to the right.

Commodore 64

The 64 version of Miami Ice is written completely in machine language and must be entered with the Commodore 64 version of the "MLX" machine language entry program found elsewhere in this issue. Be sure you read and understand the instructions for using MLX before you begin entering the data from Program 2. When you first run MLX, you'll be asked to supply a starting address and an ending address. Here are the addresses you'll need for Miami Ice:

Starting address: 0801
Ending address: 1320

After entering all the data from Program 1, be sure to save at least one copy before you exit MLX. Although the 64 version of Miami Ice is written in machine language, you start the program as if it were written in BASIC: load the program, then type RUN and press RETURN.

Plug a joystick into port 1. To steer your car safely into the parking garage and advance to the next screen, you have to enter the front of the garage without bumping into the black lines which mark its three walls. Indicators on the screen show the timer value and your current score.

There are seven screens in all. The game normally starts at screen 1, but you can begin a new game at any screen you want by moving the joystick up or down to change the screen number. This lets you skip the easier screens as you become a better player, or peek at the hardest screens while you're still a beginner.

Atari 400/800/XL/XE

The Atari version of Miami Ice (Program 3) is written largely in BASIC, but has an interrupt-driven machine language subroutine to move the car using player/missile graphics. The car itself is composed of all four players to gain more resolution and colors.

Plug a joystick into port 1. To steer your car safely into the parking garage and advance to the next screen, you have to enter the front of the garage without bumping into the black lines which mark its three walls. Indicators on the screen show the timer value and your current score.

There are seven screens in all. You'll notice that some screens have more than one route to the garage. The first time you play the game, it starts at screen 1. Subsequent games begin at the screen where the last game ended. But you can start a new game at any screen you want by moving the joystick up or down to change the screen number.

Apple II Series

The Apple version of Miami Ice is written completely in machine language and must be entered with the Apple version of the "MLX" machine language entry program found elsewhere in this issue. Be sure you read and understand the instructions for using MLX before you begin entering the data from Program 4. When you first run Apple MLX you'll be asked for a starting address and an ending address. Here are the addresses you'll need for Miami Ice:

Starting address: 1000
Ending address: 1597

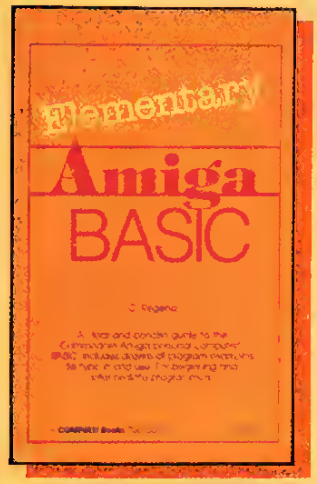
After you have typed in all the data from Program 4, be sure to save at least one copy before you exit MLX. To start MLX, enter BRUN *"filename"* (where *filename* is the name you used when you saved the Miami Ice data with MLX), then press RETURN.

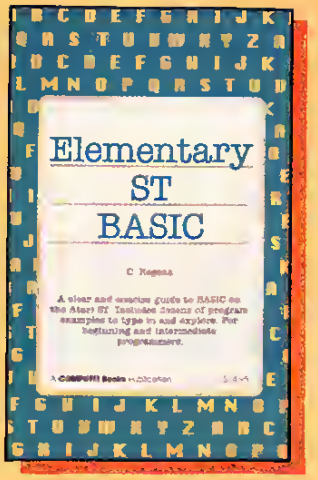
To begin playing, plug in a joystick or paddles. To reach the garage safely and advance to the next screen, you have to enter the front of the garage without bumping into any of its walls. There are seven screens in all. The game normally starts at screen 1, but you can begin a new game at any screen you want by pressing the controller button to change the screen number.

Program 1: Miami Ice For Commodore 128

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!

```
EF 10 OPEN2,8,2,"HI-SCORE,S,W"
      :CLOSE2:OPEN15,8,15:INPU
      T#15,A$,B$:IFB$<>"FILE E
      XISTS"THENCLOSE15:GOSUB7
      50
EM 20 COLOR0,16:COLOR4,11
BR 30 PRINT"{CLR}{RED}{7 DOWN}
```





BASIC programming at its best!

Two new programming guides from COMPUTE! Books.

Elementary Amiga BASIC
C. Regena
0-87455-041-6, $14.95
Disk $15.95

Elementary ST BASIC
C. Regena
0-87455-034-3, $14.95
Disk $15.95

Written by the author of the bestselling *Programmer's Reference Guide to the TI-99/4A*, these books introduce you to the new and powerful BASIC on the Amiga and Atari ST personal computers. The computers' impressive graphics, animation, and sound can be unlocked with the right commands, and BASIC is the place to start. Regena shows you how—in the clear, concise language that's made her such a popular writer.

Complete descriptions of the Amiga's and ST's BASIC commands, syntax, and organization take you from novice to full-power programming. Sample programs and subroutines, all ready to type in, are included. Plus, both books offer you working software while showing you how to write your own programs. *A disk is also available for each book which includes all the programs from the book in an easy, ready-to-load format.*

SPECIAL COMBINATION OFFER
Order the book and disk together for only $29.95!

You'll find these new programming guides and many more useful, entertaining COMPUTE! books at your local computer and book stores. Or you can order directly from COMPUTE! Books.

For the fastest service, call toll free 1-800-346-6767 (in NY 212-887-8525). Or mail the attached coupon with your payment to COMPUTE! Books, P.O. Box 5038, F.D.R. Station, New York, NY 10150.

Please add shipping and handling charges to all orders: in U.S., $2.00 per book, disk, or combination package; $5.00 per item airmail.

I want the best in BASIC programming guides. Please send me:

_____ ***Elementary ST BASIC,*** (034-3), $14.95 each
_____ ***Elementary ST BASIC Disk,*** $15.95 each
_____ ***Elementary ST BASIC*** disk and book combination, $29.95
_____ ***Elementary Amiga BASIC,*** (041-6), $14.95 each
_____ ***Elementary Amiga BASIC Disk,*** $15.95 each
_____ ***Elementary Amiga BASIC*** disk and book combination, (057-2), $29.95

Subtotal _____
NC residents add 4.5% sales tax _____
Shipping charges _____
($2.00 per item in U.S. and surface mail.
$5.00 per item airmail)
Total amount enclosed _____

ALL ORDERS MUST BE PREPAID IN U.S. FUNDS

☐ Payment enclosed (check or money order)
☐ Charge ☐ MasterCard ☐ Visa ☐ American Express

Account No. _____ Exp. Date _____
(Required)

Name _____
Address _____
City _____
State _____ Zip _____

Allow 4–6 weeks for delivery

36612111

COMPUTE! Publications, Inc.
Part of ABC Consumer Magazines Inc
One of the ABC Publishing Companies
825 7th Avenue 6th Floor New York NY 10019
Publishers of COMPUTE! COMPUTE!'s Gazette COMPUTE!'s Gazette Disk COMPUTE! Books and COMPUTE!'s Apple Applications

COMPUTE! books are available in the U.K., Europe, the Middle East, and Africa from Holt Saunders, Ltd., 1 St. Anne's Road, Eastbourne, East Sussex BN21 3UN, England and in Canada from McGraw-Hill, Ryerson Ltd., 330 Progress Ave., Scarborough, Ontario, Canada M1P 2Z5.

Lyco Computer Marketing & Consultants

ATARI

ATARI

130XE	CALL
65XE	CALL
800XL	CALL
520ST	CALL
1050 Drive	145
1027 Printer	145
850 Interface	109
SF314 Drive	229
SF354 Drive	179
1040 ST (New)	***Call***
XM301 Modem	***31***

ACTIVISION (Atari)

Hacker	15 75
Mindshadow	15 75
Ghostbusters	15 75
Great Am. Race	15 75
Music Studio	20 75
Space Shuttle	15 75

MICROLEAGUE (Atari)

Baseball	24 95
GM disk	24 95
Team disk	14 95

MICROPROSE (Atari)

Kennedy Approach	21 75
Crusade in Europe	24 75
Decision in Desert	24 75
Solo Flight	20 75
Nato Commander	20 75
Spitfire Ace	18 75
F-15 Strike Eagle	20 75
Silent Service	20 75
Conflict in Nam	24 75

BRODERBUND (Atari)

The Print Shop	28 95
Graphics Library	16 95
Graphics Library II	19 50
Graphics Library III	19 50
Bank St. Writer	42 75
Whistler's Brother	16 95
Spelunker	18 95
Stealth	18 95
Serpent's Star	24 95
Mask of the Sun	24 95

SYNAPSE (Atari)

Synfile	29 95
Syncalc	29 95
Template	14 95
Mindwheel	24 75
Brimstone	24 75

SSI (Atari)

Nam	24 75
Mechraged	34 95
Antietam	29 95
USAF	34 95

VIP TECH

VIP Professional 520ST	115
VIP LITE 520St	65
VIP Professional Amiga	129

ACTIVISION (520 St)

Borrowed Time	29 75
Music Studio	29 75
Hackler	26 75
Mindshadow	29 75

HABA (52D St)

Writer	49 95

QUICKVIEW (520 St)

Zoomracks	49 95

MARK of UNICORN (520ST)

HEX	29 95
MINCE	129 95
PC/InterComm	99 95

SUB LOGIC (Atari)

Flight Simulator II....32.75

Night Mission Pinball	20 75

apple

ACTIVISION (Apple)

Alter Ego	28 75
Little People	24 75
Mindshadow	24 75
Hackler	24 75
Gamemaker	24 75

BRODERBUND (Apple)

The Print Shop	31 50
Graphic Librarys EA	18 50
Bank St. Writer 128K	42 75
Bank St. Speller	42 75
Carmen Sandiego	22 75
Karateka	22 75
Captain Goodnight	22 75
Muppet Cruise	25 75
P.S. Companion	24 75
Science Kit	35 95

MICROPROSE (Apple)

Crusade in Europe	24 75
Decision in Desert	24 75
F-15 Strike Eagle	20 75
NATO Commander	20 75
Silent Service	20 75
Solo Flight	20 75

SSI (Apple)

Phantasie II	24 75
Wizard's Crown	24 75
Rings of Zilfin	24 75
Colonial Conquest	24 75
Battlegroup	35 75
NAM	29 75

MICROLEAGUE (Apple)

M.L. Baseball	24 95
General Mgr	24 95

COMMODORE

COMMODORE

128	CALL
C 1571 Drive	CALL
C 1902-A	CALL
C-64	CALL
C 1541 Drive	CALL
C 1670 Modem	CALL
C-64 Computer	CALL
MPS801 Printer	LOW
C 1350 Mouse	42
C 1700 128K RAM	145
C 1750 512K RAM	269
JANE	35
Perfect Writer	49
Perfect Calc	49
Perfect Filer	49

ACTIVISION (C-64/128)

Alter Ego	28 75
Hackler	18 75
Little People	20 75
Gamemaker	24 75
Borrowed Time	18 75
Space Shuttle	18 75
Music Studio	***24.75***
Mindshadow	18 75
Roadrace	18 75
Fast Tracks	22 75
Count Down	18 75
Ghostbusters	22 75

MICROLEAGUE (C-64)

Baseball	24 95
GM disk	24 95
Team disk	14 95

WICO Joysticks

15 9714 Bat Handle	16 75
50-2030 Boss	11 99
50-2002 Super 3-Way	19 99

MICROPROSE (C-64)

Kennedy Approach	21 75
Crusade in Europe	24 75
Decision in Desert	24 75
Solo Flight	20 75
Nato Commander	20 75
Spitfire Ace	18 75
F-15 Strike Eagle	20 75
Acrojet	21 75
Silent Service	21 75
Conflict in Nam	24 75
Gunship	21 75

BRODERBUND

The Print Shop	28 75
Graphics Library I, II, III	18 75
Karateka	19 75
Bank St. Writer	32 75
Lode Runner	21 75
Printshop Companion	***24 75***
Bank St. Speller	32 75
Bank St. Filer	32 75
Bank St. Mailer	32 75
Music Shop	***27.75***
Championship Loderunner	21 75

SUBLOGIC

Night Mission Pinball	20 75
Flight Simulator	***31 75***
Jet Simulator	25 95
Football	NEW
Scenery Disk EA	14 95
Set 1-6	69 95

CARDCO

Numeric Keypad	34 95
CB/5 5-slot Board (64)	49 95
CB/2 2-slot Board (64)	21 95
S More Basic Rom	49 95
Write Now 64	32 95
Mail Now 64	24 95
Spell Now 64	24 95
File Now 64	24 95
Paint Now 64	24 95
Calc Now-64	24 95
Super Printer Utility	24 95

SYNAPSE

Syncalc	29 95
Template	14 95
Loderunner Rescue	19 95
Essex	24 95
Brimstone	24 95
Mindwheel	24 95

UNISON WORLD

Print Master (Amiga)	22 75
Print Master (C 128)	22 75
Print Master (C 64)	22 75
Art Gallery	16 75

MODEMS

US ROBOTICS

Password 1200	229
Password 300	139
Courier 2400	469

HAYES

Smartmodem 300	133
Smartmodem 1200	377
Smartmodem 1200B	347
Smartmodem 2400	598

SUPRA

Supra 300 (Atari)	49 95
Supra 1200 (Atari)	199 95

DIGITAL DEVICES

Pocket Modem AT	Call
Compuserve	***18 95***

RACAL-VADIC

2400PC	549
2400PA	799
2400V	559
1200PC	329

SUPRA

1064 Modem (C-64)	49 95

COMMODORE

1670 Modem	155

ANCHOR

Volksmodem	55
Volksmodem 12	186

MONITORS

PANASONIC

DTH103 10" RGB Hi Res	395
TX12H3P 12" Color	419
TR120M8PA 12" Amber	109
TR122M9P 12" Green IBM	148
TR122MYP 12" Amber IBM	148

TEKNIKA

MJ-10 Composite	179
MJ-22	255

ZENITH

ZVM 122A Amber	75
ZVM 123G Green	75
ZVM 124 Amber IBM	129
ZVM 131 Color	275
ZVM 133 RGB	389
ZVM 135 Composite	449
ZVM 136 Hi Res Color	589
ZVM 1220	95
ZVM 1230	95
ZVM 1240	149

SAKATA

SG 1000 12" Green	99
SA 1000 12" Amber	109
SG 1500 12" Green TTL	119
SA 1500 12" Amber TTL	129
SC 100 13" Color Comp	209
SC 200 13" RGB	389
STS1 Tilt Stand	29

COMMODORE

1902 Color	CALL
1802 Color	CALL

PRINCETON GRAPHICS

MAX 12 Amber	165
HX 12 RGB	465
SR 12 RGB	595

AMDEK

300 Green	118
300 Amber	128
310 Amber IBM	155
Color 300 Audio	234
Color 500 Composite	369
Color 600	397
Color 700	495
Color 710	569

THOMSON

CM36512V1	269
CM36632	159

NEW HOURS!
Mon-Thur - 9 AM-8 PM
Fri - 9 AM-6 PM
Sat - 10 AM-6 PM

LYCO COMPUTER

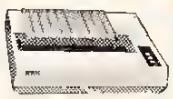
America's Mail Order Headquarters

NEW HOURS!
Mon-Thur - 9 AM-8 PM
Fri - 9 AM-6 PM
Sat - 10 AM-6 PM

Lyco Computer Marketing & Consultants

1091 . . . . . $228

SAVE ON THESE IN STOCK PRINTERS



SG-10 . . . . $205

LEGEND

808	148
1080	197
1380	224
1385	268

BROTHER

HR-15XL-P	359
HR-15XL-S	359

SILVER REED

EXP400	249
EXP500	295
EXP550	399
EXP770	749

PANASONIC

1091	228
3131 (NEW)	264
1092	325
3151	409
1080 (NEW)	209
1592 (NEW)	439

EPSON

LX80	209
FX85	333
JX80	Call
Homewriter 10	193
DX10	207
DX20	297
DX35	597
AP-80	244
HI-80	355
HS 80	298
FX-286 (NEW)	489
LQ-800 (NEW)	529
LQ-1000 (NEW)	659

SEIKOSHA

SP 1000 VC (C 64)	185
SP-1000 A Centronics	199
SP 1000 I IBM	199
SP-1000 AS RS-232	199
SP1000 AP Apple IIc	199
BP-5200 I	649
BP sheet feeder	199
SP 1000 ribbon	8 50
BP-5200 ribbon	12 50

OKIDATA

Okimate 10	179
182	214
192	348
193	563

DIABLO

D25	549
630 API	1599
630 ECS	1759
D 80 1F	2395
P 32 CO1	699
P 36	1749
C 150	999

STAR MICRONICS

SG-10	205
SG-10c	219
SG 15	367
SD-10	319
SD 15	438
SR 10	469
SR 15	578
SB 10	589
Powertype	297
NX 10 (NEW)	CALL
NB 15 (NEW)	CALL

JUKI

Juki 6100	344
RS232 Serial Board	55
6100 Tractor	119
6100 Sheet Feeder	209
Juki 6300	757

CORONA

LP300	2495
200361 Toner Cartridge	89

CITIZEN

MSP-10	255
MSP-15	355
MSP-20	337
MSP 25	495
120D	188
Premier 35	429

C. ITOH

Prowriter 8510 sp+	Call
15505 sp+	Call
Printmaster	Call

COLOR RIBBONS NOW AVAILABLE!!

DUST COVERS

Atari

520ST	11 95
130XE	6 99
800XL	6 99
1050	6 99
1025	7 99

Commodore

C128	7 99
1571/1541	6 99
1902	10 95
1702	8 99
C64/Vic20	6 99

Panasonic

1090/1091	8 99
1092	8 99
1093	9 99

Star Micronics

SG/SD10	8 99
SG/SD15	9 99
SR10	9 99
SR15	9 99

Okidata

82/92	8 99
83/93	9 99
193	9 99

DRIVES

INDUS

GT Atari	195
GT Commodore	195

COMTEL

Enhancer 2000 (C 64)	159

TYMAC

MDD-640 3½" Apple Drive 640K	269

COMMODORE

1571	CALL
1541	CALL

INTERFACING

DIGITAL DEVICES

Ape Face XLP (Atari)	49
U-Print A (Atari)	54
U-Print A 16K Buffer	74
U-Print AP 16K (Apple)	99

CARDCO

G-WIZ (C-64)	54
C/?PS (C-64)	49
C/?B (C-64)	39

XETEC

Super Graphix 64	64
Super Graphix JR 64	45

MICROBITS

MPP-1150 (Atari)	54
MPP 1150XL (Atari)	59
MicroPrint (Atari)	39

TYMAC

Connection (C-64)	55
Tackler (Apple)	49
PPC 100 (Apple)	39

MICROTEK

Dumpling GX (Apple)	59
Dumpling 16K (Apple)	89
RV 611C (Apple)	49

ORANGE MICRO

GRAPPLER+ (Apple)	85
Grappler 16K (Apple)	149
ORANGE (Apple)	59
Grappler CD (C 64)	79

DISKETTES

DENNISON

ELEPHANT 5¼ SSSD	11 99
ELEPHANT 5¼ SSDD	12 99
ELEPHANT 5¼" DSDD	14 99

SUNKYONG

SKC 5¼ SSDD	11 99
SKC 5¼' DSDD	13 99

MAXELL

5¼" MD1	13 99

VERBATIM

5¼" SSDD	13 99
5¼" DSDD	19 99

BONUS

5¼ SSDD	8 99
5¼ DSDD	12 99

IBM-PC

SSI (IBM)

Battle for Normandy	24 95
Knights of Desert	24 95
Tigers in Snow	24 95
Computer Baseball	24 95
Cartels & Cutthroats	24 95
OP Market Garden	29 95
50 Mission Crush	24 95

MICROPROSE (IBM)

F-15 Strike Eagle	20 75
Solo Flight	20 75
Silent Service	20 75
Decision in Desert	24 95
Crusade Europe	24 95

SUBLOGIC (IBM)

Jet Simulator	34 95
Scenery Disks EA	14 95
Set 1-6	69 95

LEADING EDGE

Nutshell	69 95
Nutshell Filer	149 00

ACTIVISION (IBM)

Borrowed Time	24 75
Mindshadows	24 75
Music Studio	29 95
Alter Ego	29 95

UNISON WORLD (IBM)

Printmaster	34 95
Art Gallery 1	24 95

SYNAPSE (IBM)

Synstock	64 95
Essex	26 95
Wizard of Wall St	26 95
Brimstone	26 95

QUICKVIEW (IBM)

Zoomracks	4 95

BRODERBUND (IBM)

Bank St Writer	48 95
The Print Shop	34 95
Graphics Library 1	22 95
Ancient Art of War	27 95
Champ Lode Runner	22 95
Karateka	22 95

MICROLEAGUE (IBM)

M L Baseball	24 95
General Mgr	24 95
85 Team Disk	14 95

TOLL FREE 1-800-233-8760

TO ORDER

CALL TOLL FREE 1-800-233-8760
In PA 717-494-1030
Customer Service 717-494-1670

or send order to
Lyco Computer
P O Box 5088
Jersey Shore, PA
17740

RISK FREE POLICY

In stock items shipped within 24 hours of order No deposit on C O D orders Free shipping on prepaid cash orders within the continental U S Volume discounts available PA residents add sales tax APO FPO and international orders add $5 00 plus 3% for priority mail service Advertised prices show 4% discount for cash add 4% for MasterCard or Visa Personal checks require 4 weeks clearance before shipping Ask about UPS Blue and Red label shipping All merchandise carried under manufacturer s warranty Free catalog with order All items subject to change without notice

```
      {15 RIGHT}{RVS}MIAMI ICE
      "
SD 40 PRINT"{8LU}{DOWN}
      {11 SPACES}JOYSTICK IN P
      ORT 2"
JJ 50 PRINT"{BLK}{DOWN}{BLK}
      {11 SPACES}[LEFT]
      {2 SPACES}TURN LEFT":PRI
      NT"{11 SPACES}[RIGHT] TU
      RN RIGHT"
QC 60 PRINT"{11 SPACES}[FIRE]
      {2 SPACES}ACCELERATE":PR
      INT"{YEL}{DOWN}
      {13 SPACES}READING DATA.
      .."
BD 70 GOSUB1800:PRINT"{UP}
      {BLU}{10 SPACES}PRESS BU
      TTON TO PLAY"
DB 80 IFJOY(2)<>128THEN80
8E 90 HY=3:SC=0:SN=1
XD 100 FAST:ONSNGOSUB760,1020,
       1280,1550:SLOW:PRINT"
       {HOME}"TAB(32);"{BLK}LI
       VES";HY:COLOR0,16:TM=40
       0:T=0:XE=0
KC 110 GOSUB540:
AH 120 POKE2041,62:MOVSPR2,X,Y
       :SPRITE2,1,2,0,0,0,1:PO
       KE2040,57:XE=8UMP(2)
MX 130 MOVSPR1,30#0:SPRITE1,1,
       9,0,0,0,1:SPRCOLOR1,2:M
       OVSPRI,40,65:I=4:AN=180
       :HT=135:TH=0:XE=8UMP(2)
       +BUMP(1)
8Q 140 PRINT"{HOME}{RVS}";TM;"
       {LEFT} {OFF}"
JX 150 IFJOY(2)=0THEN150
PH 160 IFJOY(2)=3THEN280
GC 170 IFJOY(2)=7THEN310
PR 180 IFJOY(2)=128THENMOVSPR1
       ,AN#1:TH=1:SOUND1,5000,
       24,2,1000,3,3
XG 190 POKE2040,53+I:IFBUMP(1)
       =3THEN490:ELSEIFBUMP(2)
       AND1THEN420
PC 200 IFTH>1THENONA8S(T-20)GO
       TO410
RC 210 T=T+1
KS 220 IFHT>180THENIFHT-180>AN
       THEN360
RS 230 IFHT>180THENIFHT-180<AN
       THEN370
KR 240 IFHT<180THENIFHT+180<AN
       THEN380
RD 250 IFHT<180THENIFHT+180>AN
       THEN390
MM 260 TM=TM-1
CC 261 IFTM<0THENTM=0
XP 262 PRINT"{HOME}{RVS}";TM;"
       {LEFT} {OFF}"
MH 270 IFJOY(2)<>3THEN300
AQ 280 AN=AN+45:IFAN>360THENAN
       =45
KD 290 I=I-1:IFI=0THENI=8:GOTO
       190
DS 300 IFJOY(2)<>7THEN340
CF 310 AN=AN-45:IFAN<0THENAN=3
       15
KE 320 IFAN=360THENAN=0
DD 330 I=I+1:IFI=9THENI=1:GOTO
       190
HH 340 IFJOY(2)=128THENSOUND1,
       5000,24,2,1000,3,3:TH=T
       H+1:T=0:IFTH>15THENTH=1
       5:GOTO190
CP 350 GOTO190
FG 360 HT=HT+((AN+(360-HT))/10
       ):MOVSPR1,HT#TH:IFHT<36
       0THEN260:ELSE:HT=0:GOTO
       390
RX 370 HT=HT-((HT-AN)/10):MOVS
       PR1,HT#TH:GOTO260
SM 380 HT=HT-((HT+(360-AN))/10
       ):MOVSPR1,HT#TH:IFHT>0T
       HEN260:ELSE:HT=360:GOTO
       370
JQ 390 HT=HT+((AN-HT)/10):MOVS
       PRI,HT#TH:GOTO260
QR 400 GOTO260
XD 410 TH=TH-1:T=0:IFTH<1THENT
       H=1:GOTO220:ELSE220
KF 420 POKE2040,63:FORDELAY=1T
       O150:NEXT:SPRITE 1,0
RJ 430 SOUND1,2000,100,0,1000,
       1,3,100
J8 440 T=0:HY=HY-1:PRINT"
       {HOME}"TA8(32)"LIVES";H
       Y:IFHY=0THEN460
CD 450 XE=BUMP(2):SLEEP2:POKE2
       040,57:MOVSPR1,33,55:XE
       =BUMP(2):GOTO130
HX 460 SLEEP2:PRINT"{9 DOWN}
       {15 RIGHT}{RVS}{BLK}GAM
       E OVER{OFF}"
PD 470 OPEN2,8,2,"HI-SCORE,S,R
       ":INPUT#2,A$,8$:CLOSE2:
       IFSC>VAL(A$)THENFORI=1T
       08:SPRITEI,0:NEXT:GOTO5
       60
HK 480 IFJOY(2)<>128THEN480:EL
       SE:XE=BUMP(2):GOTO90
AS 490 POKE53280,6:MOVSPR1,40#
       0:PLAY"QGRGRG":IFTM=0TH
       EN520
GP 492 SD=INT(2000/TM)
XG 500 FORTY=1TOTMSTEP5:PRINT"
       {HOME}{RVS}";TM-TY:PRIN
       T"{HOME}{RVS}";TAB(13);
       SC+TY:IFTM-TY<99THENIFT
       M-TY>90THENPRINT"{HOME}
       {RVS}{4 SPACES}{OFF}"
HM 510 SOUND1,3000+(SD*TY),1:N
       EXT:SC=SC+TM:PRINT"
       {HOME}{RVS}{2 SPACES}0
       {SPACE}{OFF}";TAB(13);"
       {RVS}";SC
PC 520 SLEEP1:SN=SN+1:IFSN=5TH
       ENSN=1
CB 530 GOTO100
QK 540 PRINT"{HOME}{RVS}";TM;"
       {HOME}{RVS}"TAB(13);SC:
       RETURN
XB 550 REM ********** HI SCORE
        **********
GB 560 PLAY"O4SCCFGBBAR AB":PR
       INT"{CLR}{2 DOWN}
       {11 SPACES}YOUR SCORE:
       {SPACE}";SC:A8=65:OP=0
JG 570 PRINT"{10 SPACES}CCCCCC
       CCCCCCCCCCCCCC{UP}"
PB 580 OPEN2,8,2,"HI-SCORE,S,R
       ":FORI=1TO10:INPUT#2,B$
       (I):INPUT#2,A$(I):NEXT:
       CLOSE2:SCRATCH"HI-SCORE
       "
G8 590 FORU=1TO10:IFSC>VAL(8$(
       U))THENNEXT
FX 600 U=U-1:FORE=1TOU-1:A$(E)
       =A$(E+1):B$(E)=B$(E+1):
       NEXT:B$(U)=RIGHT$(STR$(
       SC),LEN(STR$(SC))-1):A$
       (U)="---"
PB 610 TE=LEN(B$(U)):FORP=1TO6
       -TE:B$(U)="0"+B$(U):NEX
       T
FG 620 PRINT"{2 DOWN}":FORE=10
```

```
        TO2STEP-1:PRINTTA8(11);
        11-E;"{2 SPACES}";A$(E)
        ;"{3 SPACES}";8$(E):NEX
        T
SF 630  PRINTTAB(10);10;"
        {2 SPACES}";A$(1);"
        {3 SPACES}";B$(1)
EQ 640  PRINT"{HOME}{5 DOWN}":F
        ORI=1TO11-U:PRINT:NEXT:
        NM$=""
CM 650  PRINT"{UP}"TAB(16+OP);C
        HR$(AB)
GF 660  IFJOY(2)=7THENAB=AB-1:I
        FAB<65THENAB=65:GOTO650
PK 670  IFJOY(2)=3THENAB=AB+1:I
        FAB>90THENAB=90:GOTO650
8O 680  IFJOY(2)=128THENNM$=NM$
        +CHR$(AB):AB=65:OP=OP+1
        :SLEEP1:IFOP=3THEN700
XA 690  GOTO650
ER 700  A$(U)=NM$:OPEN2,8,2,"HI
        -SCORE,S,W":FORI=1TO10:
        PRINT#2,B$(I):PRINT#2,A
        $(I):NEXT:CLOSE2
GF 710  PRINT"{HOME}":FORI=1TO1
        8:PRINT:NEXT
MG 720  PRINT"{7 SPACES}PRESS B
        UTTON TO PLAY AGAIN":GO
        TO480
FM 740  REM ******* CLEAR HI-SC
        ORES *******
GH 750  SCRATCH"HI-SCORE":PRINT
        "{CLR}MAKING HI-SCORE":
        OPEN2,8,2,"HI-SCORE,S,W
        ":FORI=1TO10:PRINT#2,"Ø
        ØØØØØ":PRINT#2,"---":NE
        XT:CLOSE2:RETURN
O8 760  X=62:Y=135:COLOR4,16
KG 770  PRINT"{CLR}{RVS}{RED}
        {28 SPACES}{C3}"
JX 780  PRINT"{RVS} {OFF}
        {24 SPACES}{WHT}++{RED}
        {C3}{RVS}{9 SPACES}{C3}
        {OFF}"
KG 790  PRINT"{RVS} {OFF}
        {25 SPACES}{WHT}{Q3}
        {5 SPACES}{Q3}+{W3}{RED}
        {C3}{RVS}{3 SPACES}{C3}
        {OFF}"
AH 800  PRINT"{RVS} {OFF}
        {31 SPACES}{E3}{RVS}
        {3 SPACES}{OFF}
        {2 SPACES}{REO}{C3}{RVS}
        {2 SPACES}{OFF}"
CD 810  PRINT"{RVS} {OFF}
        {37 SPACES}{RVS}
        {2 SPACES}{OFF}"
FC 820  PRINT"{RVS} {C3}{OFF}
        {36 SPACES}{C3}{RVS}
        {OFF}"
EX 830  PRINT"{RVS}{9 SPACES}
        {C3}{OFF}{29 SPACES}
        {RVS}{2 SPACES}{OFF}
        {WHT}++++{REO}
        {3 SPACES}{C3}{RVS}
        {4 SPACES}{C3}{OFF}
        {24 SPACES}{BLK}{Q3}
        {RED}{RVS} {OFF}"
HS 840  PRINT"{RVS} {OFF}{WHT}+
        ++{RED}{5 SPACES}{WHT}+
        {RED}{C3}{RVS}
        {15 SPACES}{C3}{OFF}
        {11 SPACES}{8LK}{Q3}
        {RED}{RVS} {OFF}"
CM 850  PRINT"{RVS} {OFF}
        {2 SPACES}{WHT}+
        {5 SPACES}{Z3}++{RED}
        {RVS}{F3}{3 SPACES}{OFF}
        {WHT}{W3}{4 SPACES}+++
        {RED}{C3}{RVS}{2 SPACES}
        {OFF}{11 SPACES}{BLK}
        {Q3}{RED}{RVS} {OFF}"
HE 860  PRINT"{RVS} {OFF}
        {9 SPACES}{WHT}{Q3}++
        {RED}{RVS}{2 SPACES}
        {OFF}{WHT}+{W3}
        {5 SPACES}{Q3}++{RED}{C3}
        {RVS} {C3}{OFF}
        {10 SPACES}{D3}{RVS}
        {OFF}"
GB 870  PRINT"{RVS} {OFF}
        {10 SPACES}{WHT}{Z3}{X3}
        {RED}{RVS} {OFF}{WHT}+
        {W3}{8 SPACES}{Q3}+{REO}
        {RVS}{2 SPACES}{OFF}
        {9 SPACES}{BLK}{Q3}{RED}
        {RVS}{2 SPACES}{OFF}"
EK 880  PRINT"{RVS} {OFF}
        {12 SPACES}{RVS} {OFF}
        {WHT}{W3}{10 SPACES}{Q3}
        {RED}{C3}{RVS} {OFF}
        {9 SPACES}{BLK}{Q3}{RED}
        {RVS}{2 SPACES}{OFF}"
AJ 890  PRINT"{RVS} {OFF}
        {12 SPACES}{T3}
        {13 SPACES}{RVS} {OFF}
        {9 SPACES}{BLK}{Q3}{RED}
        {RVS}{2 SPACES}{OFF}"
PQ 900  PRINT"{RVS} {OFF}
        {26 SPACES}{RVS} {OFF}
        {9 SPACES}{BLK}{Q3}{REO}
        {RVS}{2 SPACES}{OFF}"
GE 910  PRINT"{RVS} {OFF}
        {37 SPACES}{C3}{RVS}
        {OFF}"
CM 920  PRINT"{RVS} {OFF}
        {37 SPACES}{BLK}{Q3}
        {RED}{RVS} {OFF}"
GR 930  PRINT"{RVS} {OFF}
```

```
        {37 SPACES}{BLK}{Q}
        {RED}{RVS} {OFF}"
MQ 940  PRINT"{RVS} {OFF}
        {37 SPACES}{BLK}{Q}
        {RED}{RVS} {OFF}"
EK 950  PRINT"{RVS} {OFF}{F}
        {11 SPACES}{@}
        {24 SPACES}{BLK}{Q}
        {RED}{RVS} {OFF}"
DQ 960  PRINT"{RVS}{2 SPACES}
        {OFF}{F}{10 SPACES}
        {RVS} {OFF} {WHT}{Q}
        {22 SPACES}{BLK}{Q}
        {RED}{RVS} {OFF}"
MK 970  PRINT"{RVS}{3 SPACES}
        {OFF}{10 SPACES}{RVS}
        {OFF}{WHT}+{W}
        {15 SPACES}{A}
        {6 SPACES}{RED}{D}{RVS}
        {OFF}"
SD 980  PRINT"{RVS}{3 SPACES}
        {OFF}{F}{WHT}{2 R}+{S}
        {4 SPACES}{RED}{D}{RVS}
        {OFF}{WHT}+++
        {12 SPACES}{A}+{W}
        {5 SPACES}{RED}{RVS}{V}
        {2 SPACES}{OFF}"
SJ 990  PRINT"{RVS}{4 SPACES}
        {OFF}{F}{WHT}+++{S}
        {RED} {D}{RVS}
        {3 SPACES}{C}{OFF}{WHT}
        {Q}++{2 R}+{6 SPACES}
        {Q}++++{S}{2 SPACES}
        {RED}{RVS}{V}{4 SPACES}
        {OFF}"
PQ 1000 PRINT"{RVS}{39 SPACES}
        {OFF}";:POKE2023,224:P
        OKE56295,2
DS 1010 RETURN
CC 1020 X=262:Y=142:COLOR4,3
KS 1030 PRINT"{CLR}{1}{RVS}
        {40 SPACES}{OFF}"
KQ 1040 PRINT"{1}{RVS} {OFF}
        {8 SPACES}{RVS} {OFF}
        {WHT}{Q}{1}{RVS} {OFF}
        {WHT}C{E}{X}
        {17 SPACES}{1}{RVS}{F}
        {7 SPACES}{OFF}"
XM 1050 PRINT"{RVS} {OPF}
        {8 SPACES}{RVS} {OFF}
        {WHT}{Q}{1}{RVS} {OFF}
        {27 SPACES}{RVS}{J}
        {OFF}{7 SPACES}{RED}
        {D}{1}{RVS}{3 SPACES}
        {OFF}{27 SPACES}{RVS}
        {J}{OFF}"
AB 1060 PRINT"{RVS} {OFF}
        {8 SPACES}{WHT}{M} {1}
        {RVS} {OFF}{BLK}{S}
        {26 SPACES}{1}{RVS}{J}
        {OFF}"
CG 1070 PRINT"{RVS} {OFF}
        {8 SPACES}{WHT}{M}
        {BLK}C+{X}{5 SPACES}
        {WHT}{Q}{RED}{RVS}{U}
        {OFF}{WHT}{S} {RED}
        {12 P}{5 SPACES}{1}
        {RVS}{J}{OFF}"
XC 1080 PRINT"{RVS} {OFF}
        {8 SPACES}{WHT}{M}
        {RED}{RVS} {OFF}
        {6 SPACES}{WHT}{Q}
        {RED}{RVS} {1}
        {14 SPACES}{OFF}{RED}
        {F}{4 SPACES}{1}{RVS}
        {J}{OFF}"
DB 1090 PRINT"{RVS} {OFF}
        {8 SPACES}{WHT}{M}
        {RED} {RVS} {OFF}
        {6 SPACES}{WHT}{Q}
        {RED}{RVS} {OFF} {WHT}
        {Q}{1}{RVS} {OFF}{WHT}
        {X} {Q}{1}{RVS} {OFF}
        {WHT}{X}{2 SPACES}{Q}
```

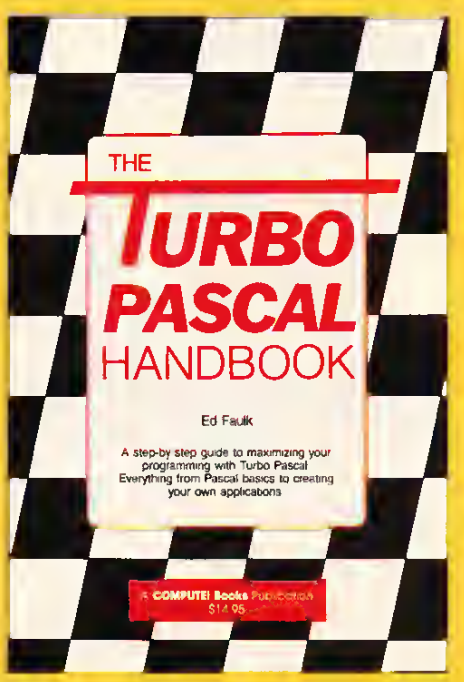
"Miami Ice," Commodore 128 version: Driving through the streets of Miami isn't easy when they're covered with a layer of ice.

```
        {1}{RVS} {OFF}{WHT}{X}
        {1}{RVS} {8LK}{L}{OFF}
        {4 SPACES}{1}{RVS}{J}
        {OFF}"
CS 1100 PRINT"{RVS} {OFP}
        {8 SPACES}{WHT}{M}
        {RED} {RVS} {OFP}
        {6 SPACES}{WHT}{Q}
        {RED}{RVS} {OFF} {WHT}
        {Q}{1}{RVS} {OFF}
        {2 SPACES}{WHT}{Q}{1}
        {RVS} {OFF}{3 SPACES}
        {WHT}{Q}{1}{2 SPACES}
        {RVS} {OFF}{BLK}{W}
        {4 SPACES}{1}{RVS}{J}
        {OFF}"
KX 1110 PRINT"{RVS}{2 SPACES}
        {OFF}{8 SPACES}{WHT}
        {Y}{RED}{RVS} {OFF}
        {6 SPACES}{WHT}{Q}
        {RED}{RVS} {OFF}
        {2 SPACES}{RED}{C}
        {2 SPACES}{WHT}{Q}
        {4 SPACES}{RED}{C}
        {2 SPACES}{1}{RVS}
        {OFF}{BLK}{W}
        {4 SPACES}{1}{RVS}{J}
        {OFF}"
XR 1120 PRINT"{RVS} {OFF}
        {9 SPACES}{RED}{RVS}
        {2 SPACES}{OFF}
        {6 SPACES}{WHT}{Q}
        {RED}{RVS} {OPF}
        {5 SPACES}{C}
        {7 SPACES}{1}{RVS}
        {OFF}{BLK}{W}
        {4 SPACES}{1}{RVS}{J}
        {OFF}"
JS 1130 PRINT"{RVS} {OFF}
        {9 SPACES}{RED}{RVS}
        {OFF}{5 SPACES}{RED}
        {D}C{RVS} {D}{OFF}
        {13 SPACES}{1}{RVS}
        {OFF}{8LK}{W}
        {4 SPACES}{1}{RVS}{J}
        {OFF}"
RQ 1140 PRINT"{RVS} {OFF}
        {9 SPACES}{RED}{RVS}
        {OFF}{6 SPACES}{WHT}
        {Q}{RED}{RVS} {OFF}{H}
        {13 SPACES}{1}{RVS}
        {OFF}{BLK}{W}
        {4 SPACES}{1}{RVS}{J}
        {OFF}"
AR 1150 PRINT"{RVS} {OFF}
        {9 SPACES}{RED}{RVS}
        {OFF}{6 SPACES}{WHT}
        {Q}{RED}{RVS} {OFF}{H}
        {13 SPACES}{1}{RVS}
        {OFF}{BLK}{W}
        {4 SPACES}{1}{RVS}{J}
        {OFF}"
RJ 1160 PRINT"{RVS} {OFF}
        {9 SPACES}{RED}{RVS}
        {OFF}{6 SPACES}{WHT}
        {Q}{RED}{RVS} {OFF}{H}
        {6 SPACES}{1}{RVS}{F}
        {7 SPACES}{OFF}{BLK}
        {W}{4 SPACES}{1}{RVS}
        {J}{OFF}"
FR 1170 PRINT"{RVS} {BLK}{D}
        {F}{OFF}{7 SPACES}
        {RED}{RVS} {OFF}
        {6 SPACES}{WHT}{Q}
        {RED}{RVS} {OFF}
        {9 SPACES}{1}{RVS}
        {BLU}{D}{F}{D}{OFF}
        {1}{RVS} {OFF}{BLK}{W}
        {4 SPACES}{1}{RVS}{J}
        {OPF}"
XJ 1180 PRINT"{RVS} {8}
        {2 SPACES}{OFF}{8LK}
        {G}{4 SPACES}{RED}{D}
        {RVS}{2 SPACES}{OFF}
        {6 SPACES}{WHT}{Q}
        {RED}{RVS} {OFF}
        {9 SPACES}{1}{RVS}{D}
        {8LU}{3 SPACES}{QFP}
        {BLK}{S}{1}{RVS} {OFF}
        {BLK}{W}{4 SPACES}{1}
        {RVS}{J}{OFF}"
8C 1190 PRINT"{RVS} {8}{D}{F}
        {OFF}{BLK}{G}
        {13 SPACES}{WHT}{Q}
        {RED}{RVS} {OFF}
        {10 SPACES}{WHT}{3 T}
        {8LK}{Z}{1}{RVS} {OFF}
        {8LK}{X}{4 SPACES}{1}
        {RVS}{J}{OFP}"
QM 1200 PRINT"{RVS} {8}
        {2 SPACES}{OFF}{BLK}
        {G}{12 SPACES}{RED}
        {RVS}{3 SPACES}{OFF}
        {14 SPACES}{1}{RVS}{D}
        {OFF}{5 SPACES}{RVS}
        {J}{OFF}"
HS 1210 PRINT"{RVS} {8}{D}{F}
        {OPF}{BLK}{G}
        {12 SPACES}{RED}{RVS}
        {3 SPACES}{OFF}
        {20 SPACES}{1}{RVS}{J}
        {OFF}"
ER 1220 PRINT"{RVS} {8}{D}{F}
        {OFF}{WHT}{S}
        {12 SPACES}{RED}{RVS}
        {SPACE}{3}Q{RED} {OFF}
        {20 SPACES}{1}{RVS}{J}
        {OFF}"
DG 1230 PRINT"{RVS} {8}{D}{F}
        {OFF}{WHT}++
        {11 SPACES}{RED}{RVS}
        {3 SPACES}{OFF}
        {20 SPACES}{1}{RVS}{J}
        {OFF}"
ER 1240 PRINT"{RVS} {8}
        {2 SPACES}{OFF}{WHT}++
        {S}{9 SPACES}{A}{RED}
        {RVS} {3}Q{RED} {OFF}
        {SPACE}{WHT} {H}
        {17 SPACES}{1}{RVS}
        {OFF}"
PK 1250 PRINT"{RVS} {8}
        {5 SPACES}{OFF}{WHT}CC
        CCCC{1}{RVS}{7 SPACES}
        {RED}{2 Y}{C}{OFF}
        {17 SPACES}{1}{RVS}
        {OFF}"
MR 1260 PRINT"{RVS}{39 SPACES}
        {OFF}";:POKE2023,224:P
        OKE56295,8
XB 1270 RETURN
8A 1280 X=280:Y=200:COLOR4,16
DK 1290 PRINT"{CLR}{GRN}{RVS}
        {40 SPACES}{OFF}"
DA 1300 PRINT"{RVS} {OFF}
        {19 SPACES}{*}{RVS}
```

Explore Pascal with

THE TURBO PASCAL HANDBOOK

from COMPUTE!.



The Turbo Pascal Handbook
Edward P. Faulk
With *The Turbo Pascal Handbook* and *Turbo Pascal* from Borland International, you'll be gently guided, step-by-step, until you're creating your own powerful applications in this impressive computer language.
$14.95 ISBN 0-87455-037-8

This information-packed book from COMPUTE! is an outstanding resource and programming guide. And it's written in COMPUTE!'s bestselling style so that even beginning programmers can quickly and easily understand all the applications.

Ask for *The Turbo Pascal Handbook* at your local computer store or bookstore. Or order directly from COMPUTE!. Call toll free 1-800-346-6767 (in NY 212-887-8525) or mail the attached coupon with your payment (plus $2.00 shipping and handling per book) to COMPUTE! Books, P.O. Box 5038, F.D.R. Station, New York, NY 10150.

Note: You'll need *Turbo Pascal* in order to use this book. The software is not included with *The Turbo Pascal Handbook.*

Yes! Send me _____ copies of *The Turbo Pascal Handbook* at $14.95 each.
My payment is enclosed.

ALL ORDERS MUST BE PREPAID IN U.S. FUNDS

Subtotal _____
NC residents add 4.5% sales tax _____
Shipping and handling _____
($2.00 per book in U.S. and surface mail, $5.00 per book airmail.)
Total enclosed _____

☐ Payment enclosed (check or money order)
☐ Charge ☐ Visa ☐ MasterCard ☐ American Express

Account No. _____ Exp. Date _____
(Required)

Name _____

Address _____

City _____ State _____ Zip _____

Please allow 4–6 weeks for delivery

36603711

COMPUTE!™ Publications, Inc. abc
One of the ABC Publishing Companies
825 7th Avenue, 6th Floor, New York, NY 10019
Publishers of COMPUTE! COMPUTE!'s Gazette COMPUTE!'s Gazette Disk COMPUTE! Books and COMPUTE!'s Apple Applications

240K Apple® Compatible Computer System

NEW

SALE

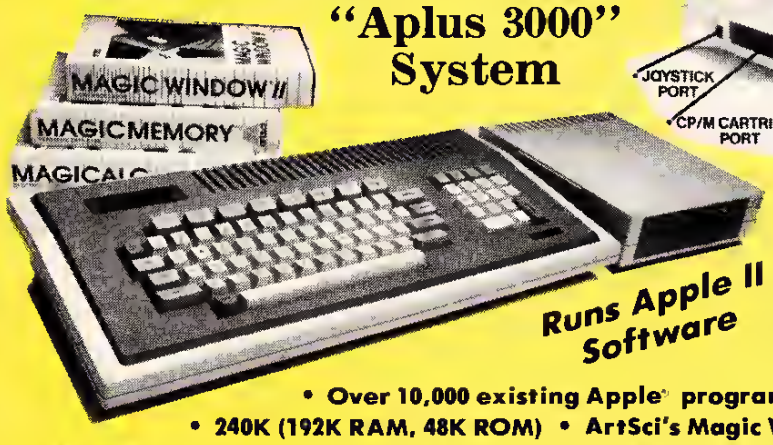
APlus 3000 computer system Includes 192K RAM, 48K ROM (32K Microsoft Basic plus 16K ROM Emulator), 160K Laser 5¼" Disk Drive (Runs Apple II Software), Magic Window Wordpracessar, MagiCalc spreadsheet, Magic Memory Database. All for only $399.00

Complete System

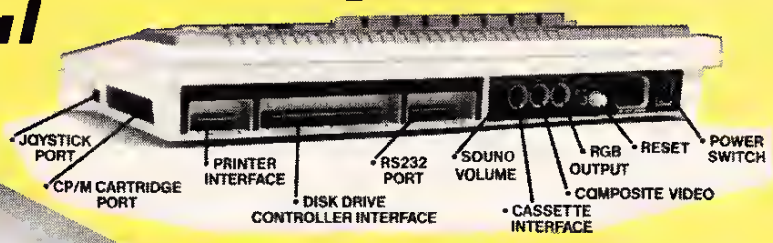
$399.00

• 15 Day Free Trial

"Aplus 3000" System



Runs Apple II Software



Double Immediate Replacement Warranty

If any af the Aplus 3000 camputer system equipment fails due ta faulty warkmonship ar material within 180 days af purchase we will REPLACE it immediately with no service charge!!

- **Over 10,000 existing Apple® programs**
- **Centranics printer interface included**
- **240K (192K RAM, 48K ROM)**
- **ArtSci's Magic Window II, Magic Memory, and MagiCalc Included**
- **160K Laser 5¼" Disk Drive (Runs Apple II software)**
- **RGB (80 columns in color) and composite included**

SPECIFICATIONS

A plus 3000 is a complete, self-contoined computer bosed on the popular 6502A microprocessor and can tap into the tremendous softwore library of Apple II. Features include 192K 8ytes RAM, 32K8 Enhanced Microsoft BASIÇ, 80 column text, 560H X 192V color grophic disploy, 81 key sculptured keyboord and high efficiency switching power supply. Also included os stondord are Centronics bus printer interface, Cassette interface, 4 chonnel sound generotor, ond 5¼" Apple Compatible Disk Drive.

- **TEXT**
  - — 40 columns X 24 rows or 80 columns X 24 rows software selectable.
  - — 5 X 7 characters in 7 X 8 matrix.
  - — Upper and lower case characters.
  - — One of Eight colors for characters/graphics and background, Red, Green, Blue, Cyan, Magenta, Yellow, Black and White.
  - — Character set with normal, inverse and flashing capabilities.
- **GRAPHICS**
  - — 280H X 192V 6 colors — Black, White, Violet, Green, Blue, Orange.
  - — 280H X 192V 8 colors bit image — Black, White, Red, Green, Blue, Cyan, Magenta, Yellow.
  - — 560H X 192V 6 colors — Block, White, Violet, Green, Blue, Orange. (High resolution color monitor required)

More Features than Apple® far less than Commodore®

Features	Aplus 3000	Apple IIe	Commodore C-128
RAM	192K	64K	128K
Runs Apple II Software	Yes	Yes	No
Function Keys	24	None	16
4 Voice, 6 Octove Sound	Yes	No	Yes
Composite Video	Yes	Yes	Yes
Disk Drive	Included	Extra Cost	Extra Cost
Numeric Keypad	Included	Extra Cost	Included
Video Cable	Included	Extra Cost	Extra Cost
RGB Color Card	Included	Extra Cost	Included
80 Column Cord	Included	Extra Cost	Included
Centronics Printer Interface	Included	Extro Cost	Extro Cost
Drive Controller	Included	Extro Cost	Included
$150 Wordprocessor (Magic Window)	Included	Extra Cost	Extro Cost
$150 Spreadsheet (MagiCalc)	Included	Extra Cost	Extra Cost
$60 Dotabase prg. (Magic Memory)	Included	Extra Cost	Extra Cost
Your Cost	**$399.00**	$1745.00	$1117.90

Super Apple Compatible Disk Drive Sale $149.95.
Quieter, Cooler, Better Disk Drives for your Apple II plus, IIe, IIc (specify when ordering). List $299.95. Sale $149.95.

15 Day Free Trial — If it doesn't meet your expectations within 15 days of receipt, just send it back to us UPS prepoid and we will refund your purchase price!!

Add $25.00 for shipping and handling!!

Enclase Cashiers Check Money Order or Personal Check Allow 14 days for delivery 2 ta 7 days for phone arders 1 day express mail! We accept Visa and MasterCard. We ship C.O.D to continental U.S. addresses anly. Add $10 more if C.O.D.

ACCESSORIES

	LIST	SALE
2nd Disk Drive	$299.95	$149.95
2 professional analog joysticks	$ 39.95	$ 24.95
Z-80 cart. allows CP/M use	$ 99.95	$ 59.95
RS232 adapter	$ 99.95	$ 59.95
R/F Modulator (TV hookup)	$ 29.95	$ 19.95
RGB cable (RGB Monitor hookup)	$ 24.95	$ 19.95
Centronics cable (for Centranics printer)	$ 34.95	$ 24.95
Technical reference manual	$ 29.95	$ 19.95
80 columns Hi-Res Green Monitor	$199.00	$ 79.95
80 column Hi-Res RGB Monitor	$399.00	$259.00

COMPUTER DIRECT
We Love Our Customers
22292 N. Pepper Rd., Borrington, Ill. 60010
312/382-5050 to order

ATARI®

152K *Lowest Price In The USA!* 152K

Computer System Sale

• Students • Word Processing • Home • Business

Limited Time
FREE Software w/ System Purchase*
Home Filing Mgr., Star Raiders, Music Composer, Defender

LOOK AT ALL YOU GET FOR ONLY **$379** SYSTEM PRICE
LIMITED QUANTITIES

	LIST PRICE	INDIVIDUAL SALE PRICE
① Atari 130XE 152K Computer	$249.00	$134.95
② Atari 1050 127K Disk Drive	299.00	159.95
③ Atari 1027 Letter Quality 20 CPS Printer	299.00	159.95
Atari Writer Plus Word Processer with Spell Checker	59.95	49.95
Atari BASIC Tutorial Manual	16.95	7.95
TOTALS	**$923.90**	**$512.75**

All connecting cables & T.V. interface included.
☆ Monitors sold separetly.

SAVE OVER $100
All 5 ONLY
$379.00
SYSTEM SALE PRICE

CALL FOR 1027 PRINTER REPLACEMENT OPTIONS

* Free software subject to substitution for other titles

Other Accessories	List	Sale	
☆ 12" Hi Resolution Green Screen Monitor	$199.00	$79.95	Add $9.95 for Connection Cables
☆ 13" Hi Resolution Color Monitor	$399.00	$149.95	Add $1D for UPS

15 DAY FREE TRIAL. We give you 15 days to try out this ATARI COMPUTER SYSTEM!! If it doesn't meet your expectations, just send it back to us prepaid and we will refund your purchase price!! **90 DAY IMMEDIATE REPLACEMENT WARRANTY.** If any of the ATARI COMPUTER SYSTEM equipment or programs fail due to faulty warkmanship or material within 90 days of purchase we will replace it IMMEDIATELY with na service charge!!

Best Prices • Over 1000 Programs and 500 Accessories Available • Best Service • One Day Express Mail • Programming Knowledge • Technical Support

Add $25.00 for shipping and handling!!
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery. 2 to 7 days for phone orders. 1 day express mail! We accept Visa and MasterCard. We ship C.O.D. to continental U.S. addresses only. Add $10 more if C.O.D., add $25 if Air Mail.

COMPUTER DIRECT

We Love Our Customers
22292 N. Pepper Rd., Barrington, Ill. 60010
312/382-5050 to order

COMMODORE 64 COMPUTER

(Order Now)

$139.95

- C128 Disks 79¢ ea.*
- Paperback Writer 64 $39.95
- 10" Comstar 10X Printer $148.00
- 13" Color Monitor $149.95

CALL BEFORE YOU ORDER

COMMODORE 64 SYSTEM SALE

Commodore 64 Plus $30.00 S&H

Com. 1541 Disk Drive
14" Color Monitor

$457

PLUS FREE $49.95 Oil Barons Adventure Program

C128 COMMODORE COMPUTER

NEW

(Order Now)

***$229.05** (SEE BELOW)

With $59.95 Timeworks Wordwriter Wordprocessor savings applied

- 340K 1571 Disk Drive $259.00
- Voice Synthesizer $39.95
- 12" Monitor $79.95

PRICES MAY BE LOWER

SPECIAL SOFTWARE COUPON

We pack a SPECIAL SOFTWARE DISCOUNT COUPON with every COMMODORE 64 COMPUTER, DISK DRIVE, PRINTER, or MONITOR we sell! This coupon allows you to SAVE OVER $250 OFF SALE PRICES!!

(Examples)

PROFESSIONAL SOFTWARE COMMODORE 64

Name	List	Sale	Coupon
PaperClip	$89.95	$34.95	$29.95
Consultant	$99.95	$49.95	$39.95
Leader Board	$39.95	$24.95	$22.95
The Print Shop	$44.95	$27.95	$26.95
Hailey's Project	$39.95	$22.95	$19.95
Practicalc (spread sheet)	$59.95	$19.95	$14.95
Voice Command Module	$79.95	$39.95	$34.95
Nine Princes in Amber	$32.95	$24.95	$21.95
Super Bowl Sunday	$35.00	$22.95	$19.95
Flip and File Disk Filer	$24.95	$14.95	$12.95
Pro Joy Stick	$19.95	$12.95	$10.00
PartyWare	$19.95	$14.95	$11.95
Dust Cover	$ 8.95	$ 6.95	$ 4.60
Financial Planner Sylvia Porter	$59.95	$38.95	$35.95
Hardball	$29.95	$18.95	$16.95
C64 Troubleshoot & Repair Guide	$24.95	$15.95	$12.95

(See over 100 coupon items in our catalog)

Write or call for Sample SPECIAL SOFTWARE COUPON!

COMMODORE 64 COMPUTER $139.95

You pay only $139.95 when you order the powerful 84K COMMODORE 64 COMPUTER! LESS the value of the SPECIAL SOFTWARE DISCOUNT COUPON we pack with your computer that allows you to SAVE OVER $250 off software sale prices!! With only $100 of savings applied, your net computer cost is $39.95!!

* C128 DOUBLE SIDED DISKS 79¢ EA.

Get these 5¼" Double Sided Floppy Disks specially designed for the Commodore 128 Computer (1571 Disk Drive) 100% Certified, ***Lifetime Warranty*** Automatic Lint Cleaning Liner included. 1 Box of 10 - $9.90 (99¢ ea.), 5 Boxes of 10 - $44.50 (89¢ ea.), 10 Boxes of 10 -$79.00 (79¢ ea.).

13" COLOR MONITOR $149.95

You pay only $149.95 when you order this 13" COLOR MONITOR. LESS the value of the SPECIAL SOFTWARE DISCOUNT COUPON we pack with your monitor that allows you to save over $250 off software sale prices!! With only $100 of savings applied, your net color monitor cost is only $49.95. (16 Colors).

Premium Quality 120-140 CPS Comstar 10X Printer $148.00

The COMSTAR 10X gives you a 10" carriage, 120-140 CPS, 9 x 9 dot matrix with double strike capability for 18 x 18 dot matrix (near letter quality), high resolution bit image (120 x 144 dot matrix), underlining, back spacing, left and right margin setting, true lower decenders with super and subscripts, prints standard, italic, block graphics and special characters. It gives you print quality and features found on printers costing twice as much!! (Centronics Parallel Interface) List $399.00 **Sale $148.00**

4 SLOT EXPANDER & 80 COLUMN BOARD $49.95

Now you program 80 COLUMNS on the screen at one time! Converts your Commodore 64 to 80 COLUMNS when you plug in the 80 COLUMN EXPANSION BOARD!! PLUS 4 slot expander! ***Limited Quantities.*** **Sale $49.95.** *Coupon $39.95*

80 COLUMNS IN COLOR PAPERBOCK WRITER 64 WORD PROCESSOR $39.95

This PAPERBOCK WRITER 64 WORD PROCESSOR is the finest available for the COMMODORE 64 computer! The ULTIMATE FOR PROFESSIONAL Word Processing, DISPLAYS 40 or 80 COLUMNS IN COLOR or black and white! Simple to operate, powerful text editing, complete cursor and insert/delete key controls line and paragraph insertion, automatic deletion, centering, margin settings and output to all printers! List $99.00 **SALE $39.95.** *Coupon $29.95.*

ATTENTION

Computer Clubs

We Offer Big Volume Discounts

CALL TODAY!

PROTECTO WARRANTY

All Protecto's products carry a minimum 90 day warranty. If anything fails within 90 days from the date of purchase simply send your product to us via United Parcel Service prepaid. We will IMMEDIATELY send you a replacement at no charge via United Parcel Service prepaid. This warranty proves once again that ***We Love Our Customers.***

* C128 COMMODORE COMPUTER $289.00

You pay only $289.00 for the C128 computer and we include the C128 Wordwriter Wordprocessor by Timeworks (Sale $59.95). Thus, your net cost for the C128 computer is only $229.05. List $349.00. **SALE $289.00**

340K 1571 COMMODORE DISK DRIVE $259.00

Double Sided, Single Disk Drive for C-128 allows you to use C-128 mode plus CPM mode. 17 times faster than 1541, plus runs all 1541 formats. List $349.00. **Sale $259.00**

SUPER AUTO DIAL MODEM $29.95

Easy to use. Just plug into your Commodore 64 computer and you're ready to transmit and receive messages. Easier to use than dialing your telephone, just push one key on your computer! Includes exclusive easy to use program for up and down loading to printer and disk drives. ***Best In U.S.A.*** List $99.00 **SALE $29.95.** *Coupon $24.95.*

VOICE SYNTHESIZER $39.95

For Commodore-64 computers. Just plug it in and you can program words and sentences, adjust volume and pitch, make talking adventure games, sound action games and customized talkies!! PLUS ($19.95 value) TEXT TO SPEECH program included FREE, just type a word and hear your computer talk — ADD SOUND TO "ZORK", SCOTT ADAMS AND OTHER ADVENTURE GAMES!! (Disk or tape.) List $89.00. **SALE $39.95**

12" MAGNAVOX (NAP) 80 COLUMN MONITOR WITH SOUND $79.95

Super High Resolution green screen monitor. 80 columns x 24 lines, easy to read, plus speaker for audio sound included. Fantastic value. List $129.00. **Sale $79.95.** (C128 cable $19.95. C64, Atari cable $9.95)

PRINTER/TYPEWRITER COMBINATION $229.95

"JUKI" Superb letter quality, daisy wheel printer/typewriter combination. Two machines in one — just a flick of the switch. 12" extra large carriage, typewriter keyboard, automatic margin control and relocate key, drop in cassette ribbon! (90 day warranty) centronics parallel or RS232 serial port built in (Specify). List $349.00. **SALE $229.95.** ***(Ltd. Qty.)***

14" RGB & COMPOSITE COLOR MONITOR $259.95

Must be used to get 80 columns in color with 80 column computers (C128 - IBM - Apple). (RGB Cable $19.95) Add $14.50 shipping. List $399.00. **SALE $259.95.**

- LOWEST PRICES • 15 DAY FREE TRIAL
- BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL

PHONE ORDERS
8 a.m. - 8 p.m. C.S.T. Weekdays
9 a.m. - 12 noon C.S.T. Saturdays

- 90 DAY FREE REPLACEMENT WARRANTY
- OVER 500 PROGRAMS • FREE CATALOGS

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6¼% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES EXCEPT CANADA. Enclose Cashier Check, Money Order or Personal Check. Allow 14 days for delivery. 2 to 7 days for phone orders, 1 day express mail! Prices & Availability subject to change without notice. VISA — MASTER CARD — C.O.D. No C.O.D. to Canada, APO-FPO

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

COLOR MONITOR SALE!!!

Sale

16 COLORS

(Premium Quality)

- **Built in Speaker & Audia**
- **Far Videa Recorders**
- **Far Small Business Computers**
- **Apple - Cammadore -Atari - Aplus 3000 -etc.**
- **One Year Free Immediate Replacement Warranty**

(Premium Quality)

- **Beautiful Calar Cantrast**
- **High Resalutian**
- **Sharp Clear Text**
- **Anti-Glare Screen**
- **40 Columns x 24 Lines**
- **Frant Panel Cantrals**

List $329.00

80 Columns **RGB** Super High Resolution

13" Color Computer Monitor

*C64/Atori composite coble $9.95

* C128 RGB/Composite 80 column coble $19.95.

Sale $159.95*

Add $14.50 Shipping

13" RGB & COMPOSITE COLOR MONITOR

Allows use of C-128 ond C64 mode - composite and 80 column RGB mode. Must be used to get 80 columns in color with 80 column computers. Speciolly designed for use with the C128's speciol composite video output, plus green screen only option switch. (odd $14.50 shipping)

List $399.00 **Sale $259.95***

12" MAGNAVOX (NAP) 80 COLUMN MONITOR

Super high resolution composite green screen monitor. 80 columns x 24 lines, eosy to reod, plus speoker for oudio sound included. Fontostic volue. Limited Quontities.

List $129.00 **Sale $79.95***

Turn Your Monitor into a TV Set Without Moving Your Computer

Elegont TV Tuner with duol UHF/VHF selector switches goes between your computer ond monitor. Includes mute, outomotic fine tuning and computer-TV selector switches. Inputs included for 300 ohm, 75 ohm, ond UHF. Con be used with coble TV ond VCR's. Fontostic Value. Limited Quontities.

List $129.95 **Sale $49.95***

15 Day Free Trial - 90 Day Immediate Replacement Warranty

• LOWEST PRICES • BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL • OVER 500 PROGRAMS • FREE CATALOGS

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6¼% tax. Monitors must be shipped to Continental U.S. Only
WE DO NOT EXPORT TO OTHER COUNTRIES. Enclose Coshier Check, Money Order or Personol Check. Allow 14 doys for delivery, 2 to 7 doys for phone orders, 1 doy express mail! Prices & Avoilobility subject to chonge without notice.
VISA — MASTER CARD — C.O.D.

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

```
        {4 SPACES}{OFF}{WHT}-
        {BLK}[13 E]{GRN}{RVS}
        {SPACE}{OFF}"
SG 1310 PRINT"{RVS} {OFF}
        {21 SPACES}[*]{RVS}
        {2 SPACES}{OFF}{WHT}-
        {13 SPACES}{GRN}{RVS}
        {SPACE}{OFF}"
JD 1320 PRINT"{RVS} {OFF}
        {22 SPACES}{RVS}
        {2 SPACES}{OFF}{WHT}-
        {13 SPACES}{GRN}{RVS}
        {SPACE}{OFF}"
MD 1330 PRINT"{RVS} {OFF}
        {13 SPACES}{RVS}£
        {OFF}{7 SPACES}[*]
        {RVS} {OFF}{WHT}-
        {13 SPACES}{GRN}{RVS}
        {SPACE}{OFF}"
XQ 1340 PRINT"{RVS}{15 SPACES}
        {OFF}£{6 SPACES}{WHT}
        UC{GRN}{RVS} {OFF}
        {WHT}K{13 SPACES}{GRN}
        {RVS} {OFF}"
CD 1350 PRINT"{RVS}{4 SPACES}
        {OFF}£{17 SPACES}
        {WHT}-{GRN}{RVS}£
        {OFF}{7 SPACES}{RVS}
        {OFF}{6 SPACES}{RVS}
        {OFF}"
CH 1360 PRINT"{RVS}{3 SPACES}
        {OFF}£{17 SPACES}
        {RVS}£{3 SPACES}{OFF}
        {7 SPACES}{RVS} {OFF}
        {6 SPACES}{RVS} {OFF}"
DG 1370 PRINT"{RVS}{2 SPACES}
        {OFF}£{17 SPACES}
        {RVS}£{3 SPACES}{OFF}
        £{7 SPACES}{RVS}
        {OFF}{6 SPACES}{RVS}
        {OFF}"
DE 1380 PRINT"{RVS}{2 SPACES}
        {OFF}{8 SPACES}[5]
        {RVS}£{GRN}
        {11 SPACES}{OFF}£
        {9 SPACES}{RVS} {OFF}
        {6 SPACES}{RVS} {OFF}
RE 1390 PRINT"{RVS} {OFF}{WHT}
        -{GRN}{7 SPACES}[5]
        {RVS}£{4 SPACES}{GRN}
        {2 SPACES}{OFF}£
        {14 SPACES}{RVS}£
        {OFF}{6 SPACES}{RVS}
        {OFF}"

JS 1400 PRINT"{RVS} {OFF}{WHT}
        -{GRN}{7 SPACES}{RVS}
        {SPACE}[5]{4 SPACES}
        {GRN} {OFF}£
        {14 SPACES}{RVS}£
        {2 SPACES}{OFF}
        {6 SPACES}{RVS} {OFF}"
HH 1410 PRINT"{RVS} {OFF}{WHT}
        -{GRN}{7 SPACES}{RVS}
        {SPACE}[5]{2 SPACES}
        {GRN}{2 SPACES}{OFF}£
        {14 SPACES}{RVS}£
        {3 SPACES}{OFF}
        {6 SPACES}{RVS} {OFF}"
QA 1420 PRINT"{RVS} {OFF}{WHT}
        -{GRN}{7 SPACES}{RVS}
        {SPACE}[5]{2 SPACES}
        {GRN} {OFF}£
        {14 SPACES}{RVS}£
        {2 SPACES}M {OFF}
        {6 SPACES}{RVS} {OFF}
BS 1430 PRINT"{RVS} {OFF}{WHT}
        -{GRN}{7 SPACES}{RVS}
        {3 SPACES}{OFF}£
        {14 SPACES}{RVS}£
        {2 SPACES}M {OFF}£
        {6 SPACES}{RVS} {OFF}"
RM 1440 PRINT"{RVS} {OFF}{WHT}
        -{GRN}{7 SPACES}{RVS}
        {2 SPACES}{OFF}£
        {14 SPACES}{RVS}£
        {2 SPACES}M {OFF}£
        {7 SPACES}{RVS} {OFF}"
RP 1450 PRINT"{RVS} {OFF}{WHT}
        -{GRN}{7 SPACES}{RVS}
        {SPACE}{OFF}£
        {14 SPACES}{RVS}£
        {2 SPACES}M {OFF}£
        {8 SPACES}{RVS} {OFF}"
CE 1460 PRINT"{RVS} {OFF}{WHT}
        -{GRN}{7 SPACES}{RVS}
        {SPACE}{OFF}
        {14 SPACES}{RVS}£
        {4 SPACES}{OFF}£
        {9 SPACES}{RVS} {OFF}"
FJ 1470 PRINT"{RVS} {OFF}{WHT}
        -{GRN}{20 SPACES}[5]
        {RVS}£ {GRN}
        {5 SPACES}{OFF}
        {10 SPACES}{RVS} {OFF}
        "
KX 1480 PRINT"{RVS} {OFF}{WHT}
        JI{GRN}{18 SPACES}[5]
        {RVS}£ {GRN}
        {3 SPACES}{OFF}{WHT}-
        {SPACE}{GRN}{RVS}
        {OFF}{10 SPACES}{RVS}
        {SPACE}{OFF}"
SH 1490 PRINT"{RVS} [5]
        {3 SPACES}{OFF}
        {17 SPACES}{RVS} {GRN}
        {3 SPACES}{OFF} {WHT}
        [Q]C{GRN}{RVS} {OFF}
        {10 SPACES}{RVS} {OFF}
        "
MX 1500 PRINT"{RVS} [5]
        {3 SPACES}{OFF}
        {16 SPACES}{GRN}{RVS}
        £{4 SPACES}{OFF}{WHT}
        C[W] {GRN}{RVS} {OFF}
        {10 SPACES}{RVS} {OFF}
        "
FF 1510 PRINT"{RVS} [5]
        {4 SPACES}{OFF}{WHT}CC
        CI{10 SPACES}{GRN}
        {RVS}£{2 SPACES}{OFF}
        {WHT} - U[W] {GRN}
        {RVS} {OFF}{10 SPACES}
        {RVS} {OFF}"
QR 1520 PRINT"{RVS} [5]
        {7 SPACES}{OFF}{WHT}-
        {8 SPACES}{GRN}[D]
        {RVS}{4 SPACES}{OFF}
        {WHT}CK {RVS} {OFF}JC
        {GRN}{RVS} {OFF}{WHT}C
        CCCCCCCCC{GRN}{RVS}
        {OFF}"
KX 1530 PRINT"{RVS}{39 SPACES}
        {OFF}";:POKE2023,224:P
        OKE56295,5
DA 1540 RETURN
CS 1550 X=110:Y=165:COLOR4,16
HK 1560 PRINT"{CLR}[5]{RVS}
        {17 SPACES}CCCCCCCC
        {3 SPACES}{OFF}[11 U]
        {RVS}[F]{OFF}"
SD 1570 PRINT"[J]{6 SPACES}[L]
        [H]{15 SPACES}[*]{RVS}
        {3 SPACES}{OFF}
        {11 SPACES}[L]"
AD 1580 PRINT"[J]{6 SPACES}
        {WHT}[C][V][5]
        {16 SPACES}[*]{RVS} -
        {OFF}{11 SPACES}[L]
K8 1590 PRINT"[J]{25 SPACES}
        [*]{RVS}-{OFF}
        {11 SPACES}[L]"
HK 1600 PRINT"[J]{26 SPACES}
        {RVS}-{OFF}{11 SPACES}
        [L]"
AE 1610 PRINT"{RVS}{16 SPACES}
        {OFF}[F]{22 SPACES}[L]
        "
EQ 1620 PRINT"{RVS}{17 SPACES}
        {OFF}[F]{15 SPACES}
        {RVS}-{OFF}{5 SPACES}
        [L]"
XF 1630 PRINT"{BLK}[6 E]+[6 E]
        +[2 E][5]{RVS}
        {3 SPACES}{OFF}[F]
        {12 SPACES}[D]{RVS}
        {OFF}{5 SPACES}[L]"
HA 1640 PRINT"[J]{5 SPACES}
        {WHT}{RVS}[I]{OFF}
        {6 SPACES}{RVS}[I]
        {OFF}{2 SPACES}[5]
        {RVS}{17 SPACES}[D]
        {OFF}{5 SPACES}[L]"
MJ 1650 PRINT"[J]{15 SPACES}
        {RVS}{3 SPACES}{OFF}
        {WHT}+++[W]{9 SPACES}
        [5][RVS][F][C]{OFF}
        {5 SPACES}[L]"
XA 1660 PRINT"[J]{15 SPACES}
        {WHT}{3 T}[W][Z]
        {10 SPACES}[5]{RVS}[V]
        {OFF}{WHT}UI
        {3 SPACES}[5][L]"
GQ 1670 PRINT"[J]{4 SPACES}
        {RED}[9 @][5]
        {18 SPACES}{RVS} {OFF}
        {BLU}[D]{RVS} {OFF}
        {WHT}-{3 SPACES}[5][L]
        [J]{4 SPACES}[3]{RVS}
        {9 SPACES}{OFF}{WHT}
        [O]{17 SPACES}[5]{RVS}
        {3 SPACES}{OFF}{WHT}-
        [5]{3 SPACES}[L][J]
        {4 SPACES}[3]{RVS} [C]
        [V][C][V][C][V][C][V]
        {OFF}{21 SPACES}{WHT}-
        [5]{3 SPACES}[L]"
KJ 1680 PRINT"[J]{5 SPACES}
        {RVS} {OFF}[7]-[5]
        {RVS} {OFF}{5 SPACES}
        [3]{RVS} C C {OFF}
        {WHT}[U]{5 SPACES}-
        {3 SPACES}[5][L]"
DX 1690 PRINT"[J]{5 SPACES}
        {RVS} {OFF}[7]-[5]
        {RVS} {OFF}{5 SPACES}
        {RVS}{14 SPACES}[*]
        {OFF}{6 SPACES}{WHT}-
        {3 SPACES}[5][L]"
SK 1700 PRINT"[J]{5 SPACES}
        {RVS} {OFF}[7]-[5]
        {RVS} {OFF}{18 SPACES}
        [*]{RVS} [*]{OFF}
        {5 SPACES}{WHT}-
        {3 SPACES}[5][L]"
SJ 1710 PRINT"[J]{5 SPACES}
        {RVS} {OFF}[7]-[5]
        {RVS} {OFF}{19 SPACES}
        [*]{RVS} [*]{OFF}
        {4 SPACES}{WHT}-
        {3 SPACES}[5][L][J]
        {5 SPACES}{RVS} [7]
        [5] {OFF}{20 SPACES}
        [*]{RVS} {OFF}
        {4 SPACES}{WHT}-
        {3 SPACES}[5][L]"
KM 1720 PRINT"[J]{5 SPACES}
        {RVS} {OFF} {RVS}
        {OFF}{21 SPACES}{RVS}
        {SPACE}{OFF}{4 SPACES}
        {BLK}[U]{3 SPACES}[5]
        [L]"
8G 1730 PRINT"[J]{5 SPACES}
        {RVS}[F]{2 SPACES}[C]
        {OFF}{20 SPACES}{RVS}
        {SPACE}{OFF}{8 SPACES}
        [L]"
FK 1740 PRINT"[J]{6 SPACES}
        {RVS}[F]{2 SPACES}
        [11 P] {OFF}{8 SPACES}
        {RVS} {OFF}{8 SPACES}
        [L]"
XQ 1750 PRINT"[J]{29 SPACES}
        {RVS} {OFF}{8 SPACES}
        [L]"
```

From the publishers of *COMPUTE!*

June 1986 *COMPUTE!* Disk

All the exciting programs from the past three issues of *COMPUTE!* are on one timesaving, error-free floppy disk that is ready to load on your Atari 400/800, XL, and XE. The June 1986 *COMPUTE! Disk* contains the entertaining and useful Atari programs from the April, May, and June 1986 issues of *COMPUTE!*.

The June 1986 *COMPUTE! Disk* costs $12.95 plus $2.00 shipping and handling and is available only from COMPUTE! Publications.

For added savings and convenience, you may also subscribe to the *COMPUTE! Disk*. At a cost of only $39.95 a year (a $12.00 savings), you'll receive four disks, one every three months. Each disk will contain all the programs for your Commodore machine from the previous three issues of *COMPUTE!*.

This is an excellent way to build your software library while you enjoy the quality programs from *COMPUTE!*.

Disks and subscriptions are available for Apple, Atari, Commodore 64 and 128, and IBM personal computers. Call for details.

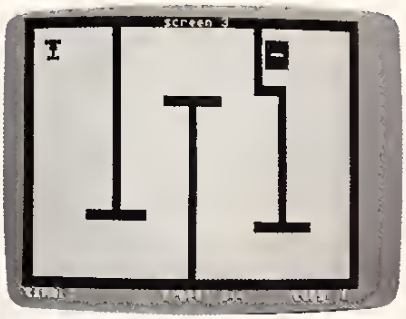
For more information or to order the June 1986 *COMPUTE! Disk*, call toll free 1-800-346-6767 (in NY 212-265-8360) or write *COMPUTE! Disk*, P.O. Box 5038, F.D.R. Station, New York, NY 10150.

COMPUTE!™ Publications, Inc. abc

Part of ABC Consumer Magazines, Inc
One of the ABC Publishing Companies
825 7th Avenue, 6th Floor New York, NY 10019

Publishers of COMPUTE! COMPUTE!'s Gazette COMPUTE!'s Gazette Disk COMPUTE! Books and COMPUTE!'s Apple Applications

```
CP 1760 PRINT"{J}{29 SPACES}
        {RVS} {DFF}{8 SPACES}
        {L}"
PX 1770 PRINT"{RVS} {29 U}
        {9 SPACES}{DFF}";:POKE
        2023,224:PDKE56295,12
CA 1780 RETURN
AQ 1800 I=3456
JS 1810 READ A:IF A=256 THEN R
        ETURN
JB 1820 POKE I,A:I=I+1:GOTD181
        0
HP 1830 DATA000,000,000,000,00
        0,000,000,000
KK 1840 DATA000,004,000,000,00
        9,000,000,040
DX 1850 DATA000,000,106,000,00
        0,018,128,016
KE 1860 DATA000,162,020,000,04
        3,133,000,011
PB 1870 DATA225,000,010,232,00
        0,018,168,000
EA 1880 DATA020,160,000,005,12
        8,000,001,000
FS 1890 DATA000,000,000,000,00
        0,000,000,000
EX 1900 DATA000,000,000,000,00
        0,000,000,000
DS 1910 DATA000,000,000,000,00
        0,000,000,000
S8 1920 DATA000,000,000,000,00
        0,000,000,000
AA 1930 DATA000,000,000,000,08
        4,000,000,084
MC 1940 DATA064,000,016,064,01
        0,168,170,171
MS 1950 DATA232,170,171,232,06
        4,010,168,064
GP 1960 DATA000,016,000,000,08
        4,000,000,084
BF 1970 DATA000,000,000,000,00
        0,000,000,000
QE 1980 DATA000,000,000,000,00
        0,000,000,000
PH 1990 DATA000,000,000,000,00
        0,000,000,000
KF 2000 DATA000,000,000,000,00
        0,000,000,000
AM 2010 DATA000,000,000,001,00
        0,000,005,128
JB 2020 DATA000,020,160,000,01
        8,232,000,011
ED 2030 DATA232,000,011,161,00
        0,042,133,000
QD 2040 DATA162,020,018,128,01
        6,106,000,000
XB 2050 DATA040,000,000,009,00
        0,000,004,000
EK 2060 DATA000,000,000,000,00
        0,000,000,000
BJ 2070 DATA000,000,000,000,00
        0,000,000,000
CP 2080 DATA000,000,000,000,00
        0,000,000,000
QP 2090 DATA000,000,004,168,06
        4,005,169,064
XG 2100 DATA005,169,064,004,18
        4,064,000,184
FH 2110 DATA000,000,184,000,00
        0,168,000,000
CG 2120 DATA168,000,000,032,00
        0,000,032,000
XP 2130 DATA000,032,000,000,03
        2,000,001,033
XX 2140 DATA000,001,169,000,00
        1,033,000,000
RX 2150 DATA000,000,000,000,00
        0,000,000,000
SS 2160 DATA000,000,000,000,00
        0,000,000,000
DD 2170 DATA000,000,000,064,00
        0,002,080,000
XJ 2180 DATA010,020,000,043,13
        2,000,043,224
JH 2190 DATA000,074,224,000,08
        2,168,000,020
E8 2200 DATA138,000,004,002,13
        2,000,000,169
FD 2210 DATA000,000,040,000,00
        0,096,000,000
JG 2220 DATA016,000,000,000,00
        0,000,000,000
KC 2230 DATA000,000,000,000,00
        0,000,000,000
MF 2240 DATA000,000,000,000,00
        0,000,000,000
EQ 2250 DATA000,000,021,000,00
        0,021,000,000
CC 2260 DATA004,000,001,042,16
        0,001,043,234
DJ 2270 DATA170,043,234,170,04
        2,160,001,004
SA 2280 DATA000,001,021,000,00
        0,021,000,000
MK 2290 DATA000,000,000,000,00
        0,000,000,000
JG 2300 DATA000,000,000,000,00
        0,000,000,000
FG 2310 DATA000,000,000,000,00
        0,000,000,000
RC 2320 DATA000,000,000,016,00
        0,000,096,000
SD 2330 DATA000,040,000,000,16
        9,004,002,132
CJ 2340 DATA020,138,000,082,16
        8,000,075,224
HF 2350 DATA000,047,160,000,04
        2,132,000,010
X8 2360 DATA020,000,002,080,00
        0,000,064,000
GR 2370 DATA000,000,000,000,00
        0,000,000,000
BQ 2380 DATA000,000,000,000,00
        0,000,000,000
GD 2390 DATA000,000,000,001,03
        3,000,001,169
CX 2400 DATA000,001,033,000,00
        0,032,000,000
KC 2410 DATA032,000,000,032,00
        0,000,032,000
HM 2420 DATA000,032,000,000,16
        8,000,000,184
QR 2430 DATA000,000,184,000,00
        4,184,064,005
HD 2440 DATA169,064,005,169,06
        4,004,168,064
CA 2450 DATA000,000,000,000,00
        0,000,000,000
RD 2460 DATA000,000,000,000,00
        0,000,000,000
JC 2470 DATA000,000,000,000,00
        0,000,000,000
HG 2480 DATA000,000,000,000,00
        0,020,000,000
SP 2490 DATA085,000,001,085,06
        4,005,085,080
QS 2500 DATA021,085,084,085,08
        5,085,106,150
JB 2510 DATA169,101,150,089,10
        1,150,089,106
AB 2520 DATA150,169,106,150,16
        9,106,150,169
BA 2530 DATA106,150,169,106,15
        0,169,106,150
JQ 2540 DATA169,106,150,169,10
        6,150,169,000
RB 2550 DATA000,000,000,000,00
        1,000,064,007
ED 2560 DATA064,064,007,064,00
        0,049,000,004
J8 2570 DATA000,010,032,000,01
        4,000,128,000
PJ 2580 DATA002,136,192,066,04
        2,131,000,168
MS 2590 DATA000,048,043,032,00
        2,168,012,016
8M 2600 DATA032,128,000,000,04
        0,003,000,064
8H 2610 DATA048,000,000,000,01
        6,016,131,000
QX 2620 DATA116,160,000,116,00
        0,000,016,000,256
```



Smash-ups are commonplace in the Commodore 64 version of "Miami Ice."

Program 2: Miami Ice For Commodore 64

Version by Kevin Mykytyn, Editorial Programmer

Please refer to the "MLX" article in this issue before entering the following listing.

```
0801:0C 08 0A 00 9E 20 32 30 64
0809:36 32 00 00 00 20 E0 0E 11
0811:20 8C 0D A9 00 A0 18 89 09
0819:A2 08 99 00 D4 88 10 F7 90
0821:20 C4 08 20 75 0A AD 01 A5
0829:DC 29 10 F0 F9 A2 00 A0 FD
0831:10 18 20 F0 FF A9 E2 A0 6F
0839:0B 20 1E AB A6 84 E8 A9 D8
0841:00 20 CD 8D A9 C8 85 F8 63
0849:A9 00 85 F9 20 D6 0D 20 15
0851:52 0A 20 95 0E A2 64 88 86
0859:D0 FD CA D0 FA AD 1F D0 55
0861:AD 1E D0 AD 01 DC 29 10 A2
0869:D0 E8 A9 40 8D 04 D4 A9 25
0871:41 8D 04 D4 20 1D 09 20 FA
0879:48 09 CE 21 13 D0 09 AD 77
0881:22 13 8D 21 13 20 95 0E 7D
0889:CE 25 13 D0 0C AD 26 13 30
0891:8D 25 13 20 16 0E 20 4F 8E
0899:0D A0 00 88 D0 FD 4C 75 6D
08A1:08 00 05 00 01 00 19 F0 81
08A9:00 1E 00 00 00 89 00 00 67
0881:0A 00 00 00 2B 00 00 05 25
08B9:F1 4F 20 CD BD A9 20 20 6C
08C1:D2 FF 60 A9 00 85 C3 85 05
08C9:C4 A9 00 85 84 A9 03 85 D6
08D1:8D A9 93 20 D2 FF A2 03 7E
08D9:8E 21 D0 E8 8E 20 D0 A2 5B
08E1:07 A0 0B 18 20 F0 FF A9 EE
08E9:EC A0 0B 20 1E AB A5 B4 9B
08F1:18 69 31 8D A7 05 A9 0A 16
08F9:20 45 0A AD 01 DC 4A B0 48
0901:0A A5 84 C9 06 F0 E7 E6 5F
0909:B4 10 E3 4A B0 08 A5 B4 40
0911:F0 DC C6 84 10 D8 4A 4A B9
0919:4A B0 D3 60 C6 F7 D0 26 DA
0921:A9 C8 85 F7 A5 F8 05 F9 7F
0929:F0 1C A5 F8 38 E9 01 85 EF
0931:F8 A5 F9 E9 00 85 F9 A2 B3
0939:18 A0 07 18 20 F0 FF A6 4D
0941:F8 A5 F9 20 BB 08 60 AD E6
0949:23 13 10 05 49 FF 18 69 E7
0951:01 85 02 AD 24 13 10 05 F2
0959:49 FF 18 69 01 18 65 02 DE
0961:8D 00 D4 AD 1F D0 29 08 46
0969:D0 14 AD 1E D0 85 02 29 4A
0971:0A C9 0A F0 09 A5 02 29 57
0979:0C C9 0C F0 3A 60 A9 40 7B
```

```
0981:8D 04 D4 A9 80 8D 12 D4 C3
0989:A9 81 8D 12 D4 A9 08 8D 8E
0991:20 13 A9 05 20 45 0A AD D5
0999:20 13 C9 0D F0 06 EE 20 28
09A1:13 4C 93 09 A9 07 8D 15 EC
09A9:D0 A9 64 20 45 0A C6 BD BA
09B1:F0 5A 68 68 4C 4D 08 A9 B7
0989:40 8D 04 D4 A9 00 85 02 77
09C1:A5 F8 05 F9 F0 32 A5 F8 89
09C9:38 E9 01 85 F8 A5 F9 E9 27
09D1:00 85 F9 A5 B4 85 03 E6 87
09D9:C3 D0 02 E6 C4 20 52 0A 06
09E1:E6 02 A5 02 8D 01 D4 A9 80
09E9:40 8D 04 D4 A9 41 8D 04 BE
09F1:D4 C6 03 10 E2 4C C1 09 56
09F9:A9 40 8D 04 D4 E6 84 A9 38
0A01:64 20 45 0A A9 05 8D 01 16
0A09:D4 4C 24 08 20 52 0A A9 A7
0A11:00 8D 15 D0 A2 0C A0 05 C3
0A19:18 20 F0 FF A9 9E A0 0B 73
0A21:20 1E AB A2 0B 20 38 0A BF
0A29:A2 0D 20 38 0A AD 01 DC 3F
0A31:29 10 D0 F9 4C 21 08 A0 2F
0A39:05 18 20 F0 FF A9 C1 A0 B3
0A41:0B 4C 1E AB 85 02 A9 00 F3
0A49:85 A2 A5 A2 C5 02 D0 FA 7A
0A51:60 A9 9F 20 D2 FF A2 18 E9
0A59:A0 17 18 20 F0 FF A6 C3 21
0A61:A5 C4 20 88 08 A2 18 A0 D4
0A69:26 18 20 F0 FF A6 BD A9 69
0A71:00 4C CD 8D A9 93 20 D2 DC
0A79:FF A9 01 8D 21 D0 A9 03 93
0A81:8D 20 D0 A2 18 A0 00 18 04
0A89:20 F0 FF A9 71 A0 0B 20 C8
0A91:1E AB A0 27 A9 A0 99 00 29
0A99:04 99 98 07 A9 04 99 00 2A
0AA1:D8 99 98 DB 88 10 ED A9 63
0AA9:00 85 F8 85 FD A9 04 85 1B
0AB1:FC A9 D8 85 FE A2 18 A0 75
0AB9:00 A9 A0 91 FB A0 27 91 A7
0AC1:F8 A9 04 91 FD A0 00 91 DB
0AC9:FD A5 F8 18 69 28 85 FB 3A
0AD1:A5 FC 69 00 85 FC A5 FD 8E
0AD9:18 69 28 85 FD A5 FE 69 9F
0AE1:00 85 FE CA D0 D1 A9 00 05
0AE9:85 FB A9 04 85 FC A6 84 57
0AF1:E0 07 90 02 A2 06 BD 09 1C
0AF9:0D 0A AA BD F8 0C 85 03 66
0B01:BD FC 0C 85 04 A0 FF C8 7A
0809:81 03 D0 19 C8 B1 03 8D 05
0B11:02 D0 8D 04 D0 8D 00 D0 DB
0B19:C8 B1 03 8D 03 D0 8D 05 B4
0B21:D0 8D 01 D0 60 84 02 48 91
0B29:48 29 1F 8D 28 13 68 4A 13
0B31:4A 4A 4A 29 06 AA A0 00 F6
0B39:68 10 02 A0 A0 8C 29 13 6E
0841:A5 FB 18 7D 37 0C 85 FB F4
0B49:A5 FC 7D 38 0C 85 FC A0 B5
0851:00 AD 29 13 F0 11 91 FB 14
0B59:A5 FB 85 FD A5 FC 18 69 8C
0861:D4 85 FE A9 04 91 FD A4 C4
0B69:02 CE 28 13 D0 D2 F0 97 B5
0B71:12 9F 54 49 4D 45 52 3A F5
0879:20 20 20 20 20 20 20 20 8F
0B81:20 20 53 43 4F 52 45 3A D6
0B89:20 20 20 20 20 20 20 20 9F
0B91:20 20 43 41 52 53 3A 20 B0
0B99:20 9D 94 20 00 12 9F 20 63
0BA1:92 47 41 4D 45 20 4F 56 6F
0BA9:45 52 20 2D 20 50 52 45 F9
0BB1:53 53 20 46 49 52 45 42 0F
0BB9:55 54 54 4F 4E 12 20 00 0A
0BC1:12 9F 20 20 20 20 20 20 B0
0BC9:20 20 20 20 20 20 20 20 DF
0BD1:20 20 20 20 20 20 20 20 E7
0BD9:20 20 20 20 20 20 20 20 EF
0BE1:00 9C 12 53 43 52 45 45 C9
0BE9:4E 20 00 92 9C 4D 20 49 FB
0BF1:20 41 20 4D 20 49 20 20 C7
0BF9:20 49 20 43 20 45 11 11 F3
0C01:11 9D 9D 9D 9D 9D 9D 9D D2
0C09:9D 9D 9D 9D 9D 1F 53 43 38
0C11:52 45 45 4E 20 31 11 11 2A
0C19:11 9D 9D 9D 9D 9D 9D 9D EA
0C21:9D 9D 9D 9D 9D 50 52 45 15
0C29:53 53 20 46 49 52 45 42 88
0C31:55 54 54 4F 4E 00 D8 FF AC
0C39:FF FF 28 00 01 00 4B FF F4
0C41:86 CD 00 37 B4 6A D1 A3 1A
0C49:E6 66 C6 90 A3 E6 65 86 5A
0C51:C5 E3 CC A3 E6 00 FB 42 8A
0C59:45 EC 47 25 EE 8B 6A 43 48
0C61:CD AA C2 02 84 45 65 00 09
0C69:2D C8 68 C3 45 A7 EB C3 F7
0C71:83 E4 83 C3 E6 C3 83 E2 61
0C79:A1 04 83 47 E6 83 C3 E3 F8
0C81:C3 45 E5 AB 83 C6 83 AA 2D
0C89:83 C6 83 AA 83 C6 45 65 57
0C91:83 00 FA 8C 66 C6 A1 E3 0C
0C99:A2 C8 E1 27 E2 A1 65 C4 11
0CA1:45 24 E1 63 05 EA C2 82 83
0CA9:EA 81 C1 E5 46 81 05 E3 54
0CB1:06 E4 A1 C2 82 AD 81 C1 F5
0C89:A6 C2 82 A4 84 E1 A1 82 E1
0CC1:A1 E1 81 68 05 C3 A1 E1 36
0CC9:C3 EA C1 81 E2 06 C1 00 81
0CD1:DC 46 65 D3 E9 64 E2 81 FB
0CD9:C1 EE 8E A2 24 AA 24 A4 43
0CE1:C9 E8 64 EC 00 73 89 46 9B
0CE9:F2 82 E1 C2 EC C5 E2 C1 8A
0CF1:A2 C4 AC C2 A1 82 B1 00 C8
0CF9:2D C8 3F 0C 46 0C 59 0C A4
0D01:6B 0C 95 0C D3 0C E8 0C F3
0D09:00 06 01 05 02 04 03 CE 0A
0D11:27 13 D0 39 A9 07 8D 27 DC
0D19:13 CE 2A 13 D0 0F A9 4B 48
0D21:8D 2A 13 AD 26 13 C9 07 E1
0D29:F0 03 EE 26 13 AD 23 13 65
0D31:F0 08 10 06 EE 23 13 4C 5F
0D39:3E 0D CE 23 13 AD 24 13 6C
0D41:F0 08 10 06 EE 24 13 4C 73
0D49:4E 0D CE 24 13 60 AE 20 81
0D51:13 AD 23 13 18 7D 77 0D A8
0D59:C9 46 90 04 C9 B9 90 03 95
0D61:8D 23 13 AD 24 13 18 7D 63
0D69:7F 0D C9 46 90 04 C9 89 06
0D71:90 03 8D 24 13 60 01 00 A4
0D79:FF FF FF 00 01 01 FF FF 9F
0D81:FF 00 01 01 01 00 A9 01 28
0D89:8D 19 D0 AD 1B 13 8D 06 EB
0D91:D0 AD 1E 13 8D 07 D0 AD 4C
0D99:1C 13 0A 0A 0A 8D 10 D0 DF
0DA1:AD 20 13 18 69 D4 8D FB 34
0DA9:07 A9 FA 8D 12 D0 AD 0D 26
0DB1:DC 29 01 F0 03 4C 31 EA 4A
0DB9:4C 8C FE A9 1B 8D 11 D0 A5
0DC1:A9 7F 8D 0D DC A9 87 8D 3D
0DC9:14 03 A9 0D 8D 15 03 A9 25
0DD1:81 8D 1A D0 60 A9 00 8D 97
0DD9:1C 13 8D 23 13 8D 24 13 D4
0DE1:A9 26 8D 1B 13 A9 3C 8D 03
0DE9:1E 13 A9 07 8D 20 13 A5 36
0DF1:A2 C5 A2 F0 FC A9 0F 8D 6C
0DF9:15 D0 A9 64 8D 21 13 8D F2
0E01:22 13 A9 07 8D 25 13 8D 4D
0E09:26 13 A9 07 8D 27 13 A9 78
0E11:37 8D 2A 13 60 20 10 0D 53
0E19:AD 23 13 30 1A 18 6D 1A 60
0E21:13 8D 1A 13 AD 1B 13 69 08
0E29:00 8D 1B 13 AD 1C 13 69 AA
0E31:00 8D 1C 13 4C 57 0E 49 8A
0E39:FF 69 01 85 02 AD 1A 13 36
0E41:38 E5 02 8D 1A 13 AD 1B 9F
0E49:13 E9 00 8D 1B 13 AD 1C DE
0E51:13 E9 00 8D 1C 13 AD 24 F6
0E59:13 30 18 18 6D 1D 13 8D 23
0E61:1D 13 AD 1E 13 69 00 8D 34
0E69:1E 13 AD 1F 13 69 00 8D CC
0E71:1F 13 60 49 FF 18 69 01 B6
0E79:85 02 AD 1D 13 38 E5 02 A7
0E81:8D 1D 13 AD 1E 13 E9 00 F9
0E89:8D 1E 13 AD 1F 13 E9 00 4A
0E91:8D 1F 13 60 AD 01 DC 4A 1A
0E99:4A 4A B0 12 20 D3 0E EE FF
0EA1:20 13 AE 20 13 E0 08 D0 67
0EA9:05 A2 00 8E 20 13 4A B0 6C
0EB1:0D 20 D3 0E CE 20 13 10 E4
0EB9:05 A2 07 8E 20 13 4A B0 5D
0EC1:10 AD 26 13 C9 02 F0 09 88
0EC9:CE 26 13 AD 26 13 8D 25 D1
0ED1:13 60 48 A9 80 8D 0B D4 58
0ED9:A9 81 8D 0B D4 68 60 A0 37
0EE1:00 B9 5A 0F 99 00 35 B9 99
0EE9:5A 10 99 00 36 89 5A 11 C8
0EF1:99 00 37 89 5A 12 99 00 AB
0EF9:38 A9 00 99 00 39 88 D0 FC
0F01:E0 A0 3F B9 DA 12 99 80 0E
0F09:39 88 10 F7 A0 02 A9 FF C7
0F11:99 00 39 99 3C 39 88 10 A4
0F19:F7 A0 36 A9 80 99 03 39 66
0F21:88 88 88 10 F8 A9 0C 8D C8
0F29:5C 39 8D 62 39 A2 E4 8E 48
0F31:F9 07 E8 8E FA 07 E8 8E 68
0F39:F8 07 A0 03 89 56 0F 99 88
0F41:27 D0 88 10 F7 A9 08 8D 3D
0F49:1C D0 A9 00 8D 25 D0 A9 28
0F51:07 8D 26 D0 60 02 00 02 35
0F59:02 00 00 00 00 00 00 00 78
0F61:00 00 00 00 10 00 00 60 60
0F69:00 00 28 00 00 A9 04 02 3D
0F71:84 14 8A 00 52 A8 00 4B A8
0F79:E0 00 2F A0 00 2A 84 00 A9
0F81:0A 14 00 02 50 00 00 40 8C
0F89:00 00 00 00 00 00 00 00 A7
0F91:00 00 00 00 00 00 00 00 AF
0F99:00 00 00 00 01 21 00 01 45
0FA1:A9 00 01 21 00 00 20 00 07
0FA9:00 20 00 00 20 00 00 20 F0
0FB1:00 00 20 00 00 A8 00 00 76
0FB9:88 00 00 B8 00 04 88 40 81
0FC1:05 A9 40 05 A9 40 04 A8 24
0FC9:40 00 00 00 00 00 00 00 08
0FD1:00 00 00 00 00 00 00 00 EF
0FD9:00 00 00 00 00 00 00 00 F7
0FE1:00 00 04 00 00 09 00 00 A4
0FE9:28 00 00 6A 00 00 12 80 67
0FF1:10 00 A2 14 00 2B 85 00 65
0FF9:08 E1 00 0A E8 00 12 A8 CA
1001:00 14 A0 00 05 80 00 01 65
1009:00 00 00 00 00 00 00 00 29
1011:00 00 00 00 00 00 00 00 31
1019:00 00 00 00 00 00 00 00 39
1021:00 00 00 00 00 00 00 00 41
1029:00 00 00 00 00 54 00 00 9A
1031:54 40 00 10 40 0A A8 AA B2
1039:AB E8 AA AB E8 40 0A A8 7E
1041:40 00 10 00 00 54 00 00 D4
1049:54 00 00 00 00 00 00 00 93
1051:00 00 00 00 00 00 00 00 71
1059:00 00 00 00 00 00 00 00 79
1061:00 00 00 00 00 00 00 00 81
1069:00 00 00 00 01 00 00 05 96
1071:80 00 14 A0 00 12 E8 00 78
1079:0B E8 00 0B A1 00 2A 85 F0
1081:00 A2 14 12 80 10 6A 00 07
1089:00 28 00 00 09 00 00 04 FF
1091:00 00 00 00 00 00 00 00 B1
1099:00 00 00 00 00 00 00 00 B9
10A1:00 00 00 00 00 00 00 00 C1
10A9:00 00 00 04 A8 40 05 A9 04
10B1:40 05 A9 40 04 B8 40 00 EF
10B9:B8 00 00 B8 00 00 A8 00 13
10C1:00 A8 00 00 20 00 00 20 2D
10C9:00 00 20 00 00 20 00 01 6F
10D1:21 00 01 A9 00 01 21 00 83
10D9:00 00 00 00 00 00 00 00 F9
10E1:00 00 00 00 00 00 00 00 02
10E9:00 00 00 00 40 00 02 50 60
10F1:00 0A 14 00 2B 84 00 2B AD
10F9:E0 00 4A E0 00 52 A8 00 7C
1101:14 8A 00 04 02 84 00 00 32
1109:A9 00 00 28 00 00 60 00 43
1111:00 10 00 00 00 00 00 00 37
1119:00 00 00 00 00 00 00 00 3B
1121:00 00 00 00 00 00 00 00 43
1129:00 00 00 15 00 00 15 00 C6
1131:00 04 00 01 2A A0 01 28 65
1139:EA AA 2B EA AA 2A A0 01 CF
1141:04 00 01 15 00 00 15 00 01
1149:00 00 00 00 00 00 00 00 68
1151:00 00 00 00 00 00 00 00 73
1159:00 00 00 00 00 00 00 00 7B
1161:01 00 00 07 40 10 07 40 05
1169:10 01 00 01 30 28 08 80 96
1171:38 02 80 C0 04 A2 03 00 FC
```

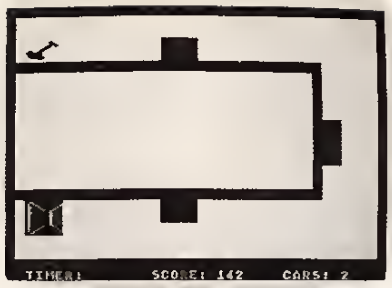
```
1179:98 01 0C AB 01 02 A8 80 46
1181:04 22 80 03 00 60 0C 10 1E
1189:00 28 C0 40 20 01 D0 C0 39
1191:01 D0 00 00 40 00 00 00 6A
1199:00 00 00 00 00 01 00 40 FF
11A1:07 40 40 07 40 00 31 00 34
11A9:04 00 0A 20 00 0E 00 80 C9
1181:00 02 88 C0 42 2A 83 00 33
11B9:A8 00 30 2B 20 02 A8 0C 4F
11C1:10 20 80 00 00 28 03 00 AA
11C9:40 30 00 00 00 10 10 83 FB
11D1:00 74 A0 00 74 00 00 10 D8
11D9:00 00 07 40 40 01 00 00 E6
11E1:00 00 04 30 00 00 00 00 87
11E9:80 00 00 02 80 00 00 8A FA
11F1:80 00 2A 0C 42 08 03 02 94
11F9:88 00 C0 20 E0 CA 28 03 81
1201:D0 A8 00 02 00 28 30 20 F8
1209:02 0C 00 40 C0 00 00 00 3B
1211:10 01 03 00 07 80 00 07 1F
1219:00 00 01 00 10 30 00 00 9E
1221:00 00 00 00 00 02 88 00 5E
1229:02 20 00 00 32 A0 00 0A 74
1231:03 00 00 80 00 82 00 02 EA
1239:A8 00 20 08 E0 3A 00 00 26
1241:28 02 08 40 02 02 00 20 37
1249:00 00 00 00 30 00 10 00 0F
1251:00 00 00 10 00 03 00 01 83
1259:00 40 30 00 00 00 00 08 9B
1261:00 00 0A 00 00 00 08 00 D6
1269:00 30 2A 00 00 A0 00 00 61
1271:00 0A 00 20 02 00 80 00 2B
1279:08 A0 20 00 88 CC 00 00 C6
1281:28 00 00 20 00 02 00 02 C5
1289:00 00 22 00 00 00 00 C0 B2
1291:00 00 00 00 01 00 10 00 DD
1299:00 08 30 00 28 00 00 00 07
12A1:00 00 00 02 00 00 0C 00 FD
12A9:00 00 0A A0 00 00 80 00 1A
12B1:00 00 00 02 00 00 08 00 06
1289:00 0A 80 00 08 30 00 20 A9
12C1:00 00 00 00 00 00 00 00 E5
12C9:00 00 00 00 00 00 00 00 ED
12D1:00 00 08 02 01 00 12 00 43
12D9:00 00 00 00 3F FF FC 5F 51
12E1:FF FA 5F FF FA 6F FF F6 3D
12E9:6F FF F6 77 FF EE 77 FF C6
12F1:EE 78 FF DE 78 F3 DE 7C 40
12F9:00 3E 78 F3 DE 78 FF DE 20
1301:77 FF EE 77 FF EE 6F FF D2
1309:F6 6F FF F6 5F FF FA 5F 46
1311:FF FA 3F FF FC 00 00 00 C5
1319:FF 00 00 00 00 00 00 00 3F
```

Program 3: Miami Ice For Atari

Version by Kevin Mykytyn, Editorial Programmer

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
BP 10  GOSUB 350:DIM B$(6),O$
       (1),ORAW$(1),GAR$(22)
FN 20  LV=1:POKE 82,0:GAR$(1,
       1)=CHR$(34):GAR$(2,22)
       ="#$%{DDWN}{4 LEFT}&'(
       ){DOWN}{4 LEFT}*+,-"
KE 30  GRAPHICS 17:POKE 559,6
       2:BETCDLDR 4,5,12:SETC
       DLDR 0,0,0:SETCOLOR 2,
       7,6:CARS=3:SCORE=0
PN 40  IF BTRIG(0)=0 THEN 40
AF 50  POSITION 5,5:PRINT #6;
       "MIAMI ICE":POSITIDN 5
       ,9:PRINT #6;"{DOWN}{4 LEFT}"
       :POSITION 2,13:PRINT #
       6;"PRESS FIREBUTTON"
CI 60  PDSITION 12,9:PRINT #6
       ;CHR$(LV+176):FOR TO=1
        TD 200:NEXT TD
```

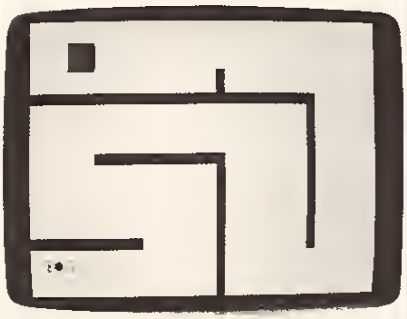


The sleek car in the Atari version of "Miami Ice" is composed of four player/missile graphics shapes.

```
BE 70  IF STICK(0)=14 AND LV<
       7 THEN LV=LV+1:GOTO 60
BC 80  IF STICK(0)=13 AND LV>
       1 THEN LV=LV-1:GOTO 60
JI 90  IF STRIG(0) THEN 70
BE 100 POKE 752,1:GOSUB 330:
       POKE 756,CHBAS:PDKE 5
       4279,CHBAS:POKE 559,6
       2:POKE 53277,3
ID 110 POKE 709,74:POKE 710,
       0:POKE 708,0:POKE 712
       ,15:A=0:GOSUB 210
MP 120 FOR A=1 TO 21:POSITIO
       N 0,A:PRINT "!
       {38 SPACES}!":NEXT A:G
       OSUB 210
JM 130 GOSUB 310:GOSUB 400:T
       I=200
KK 140 GOSUB 320
DO 150 POKE 28,1:A=USR(29195
       ):A=USR(291B4):POKE 5
       327B,255:POKE 209,0
PB 160 IF STRIG(0) THEN 160
JP 165 POKE 28,0:IF PEEK(209
       )>2 THEN 230
OF 170 ON PEEK(209) GDTO 230
       ,280:SOUNO 1,70-(PEEK
       (29)-PEEK(30)),6,4:GD
       SUB 220:GOTO 165
ML 180 IF STICK(0)=11 THEN S
       P=SP+1-B$(SP=7)
JK 190 IF STICK(0)=7 THEN SP
       =SP-1+B$(SP=0)
BP 200 IF STRIG(0)=0 THEN V=
       V+(V<10)
CN 210 POSITION 0,A:PRINT "!
       !!!!!!!!!!!!!!!!!!!!
       !!!!!!!!!!!!!!!!!!!";:
       RETURN
DK 220 TI=TI-0.1*(TI>0):PDSI
       TIDN 9,23:PRINT INT(T
       I);" ";:RETURN
LP 230 POKE 28,1:FDR A=8 TD
       11 BTEP 0.5:PDKE 205,
       A:SDUNO 1,100,8,23-A:
       NEXT A:FDR A=11 TO 0
       STEP -0.1:SDUNO 1,100
       ,8,A
JN 240 NEXT A:CARS=CARS-1:IF
        CARS<>0 THEN 140
DI 250 GOSUB 320:GOSUB 310:P
       DBITIDN 15,10:PRINT "
       GAME DVER":POSITIDN 1
       2,12:PRINT "PRESS FIR
       EBUTTDN"
PI 260 IF STRIG(0) THEN 260
OF 270 GDTO 30
LE 280 POKE 28,1:IF TI THEN
       FDR A=1 TD INT(TI):SC
       ORE=SCDRE+LV:GDSUB 30
       0:SOUNO 1,200-A,10,10
       :NEXT A
HE 290 LV=LV+(LV<7):POKE 205
       ,11:SDUNO 1,0,0,0:POK
       E 31,1:GOTO 130
BE 300 PDSITION 22,23:PRINT
       SCORE;" ";:RETURN
JH 310 FOR A=1 TO 21:POSITID
       N 1,A:PRINT "
       {38 SPACES}";:NEXT A:R
       ETURN
CP 320 PDSITIDN 0,23:PRINT "
          TIMER:{7 SPACES}SCO
       RE:{8 SPACES}CARS: ";
       CARS;" ";:GDSUB 300:R
       ETURN
AH 330 GRAPHICS 0:POKE 559,6
       2:OL=PEEK(560)+256*PE
       EK(561):POKE OL+3,68:
       FOR I=OL+6 TO DL+27:P
       DKE I,4:NEXT I:POKE I
       ,2
CB 340 I=I+1:POKE I,65:POKE
       I+1,0:POKE I+2,OL/256
       :RETURN
IL 350 CHBAS=120:PDKE 106,CH
       BAS-8:GRAPHICS 0:PDKE
        752,1:POSITIDN 14,10
       :PRINT "PLEASE WAIT":
       CHSET=CHBAS*256
LF 360 GOSUB 660
HC 370 FOR A=0 TO 1023:POKE
       CHSET+A,PEEK(57344+A)
       :NEXT A:RESTORE 540:F
       DR A=CHSET+16 TO CHSE
       T+111:READ B
NK 380 POKE A,B:NEXT A:POKE
       756,CHBAS:FOR A=CHSET
       +8 TO CHSET+15:POKE A
       ,85:NEXT A:PDKE 54279
       ,CHBAS:POKE 206,CHBAS
       +4
EK 390 POKE 559,62:POKE 623,
       4:POKE 704,55:POKE 70
       6,55:POKE 705,0:POKE
       707,0:POKE 53277,3:RE
       TURN
GK 400 SC=PEEK(88)+256*PEEK(
       89):RESTORE 440:RESTD
       RE 440+LV*10
JF 410 READ B$:IF B$="END" T
       HEN READ X,Y:PDSITIDN
        X,Y:PDKE 752,1:PRINT
       GAR$:RETURN
KJ 420 O$=B$(1,1):ORAW$=B$(2
       ,2):LENGTH=VAL(B$(3,L
       EN(B$)))
GG 430 FOR A=1 TO LENGTH:SC=
       SC-40*(O$="U")+40*(O$
       ="D")+(O$="R")-(O$="L
       "):IF ORAW$="Y" THEN
       POKE SC,1
CA 440 NEXT A:GOTO 410
BI 450 DATA DN11,RY33,UY7,DY
       14,END,2,17
JF 460 DATA DN5,RY16,UY2,RY3
       ,DY1,LY2,DY1,RY15,DY5
       ,RY2,DY3,LY1,UY2,LY1,
       DY5,LY13,DY2,LY3,UY1,
       RY2,UY1,LY17,END,2,17
DN 470 DATA RN10,DY1B,LY4,RY
       7,UN12,RN2,RY7,LY3,DY
       16,RN5,UN4,RY10,LY3,U
       Y12,LY3,UY5,END,30,2
MA 480 DATA RN7,DY1B,RY7,RN4
       ,RY16,UY14,LY12,LN4,L
       Y6,DY9,RY7,RN4,RY10,E
       ND,14,9
AD 490 DATA DN5,RY9,RN5,UN5,
       DY10,LYB,DN5,LN6,RY11
       ,DY3,UY3,RYB,UY10,DY1
       0,RY14,UY10,LNB,DN6,U
       Y10,END,2,16
IJ 500 DATA RN6,DY16,RY5,DY2
       ,UY2,RY6,DN7,UY3,UN4,
       RY5,DY2,UY2,RY5,DN7,U
       Y3,UN4,RY7,UN6,RN5,LY
```

```
      16,UY5,DY5,LY10
NN 505 DATA UY5,RN5,UY4,RN11
      ,UN1,DY5,RY5,END,30,1
NK 510 DATA DN6,RY6,DY3,UY3,
      RY4,UN6,DY2,DN4,RY8,U
      Y2,DY2,RY1,DY3,UY3,RY
      5,UN6,DY2,DN4,RY5,UY2
      ,DY2,RY4,DY3
AN 520 DATA DN5,RN6,LY9,LY4,
      UY3,DY3,LY8,DY4,UY4,L
      Y5,UY3,DY3,LY7,DY4,RN
      6,UN1,DY5,RN13,DN1,UY
      5,END,30,15
BP 540 DATA 85,122,110,107,1
      06,106,106,106
IK 550 DATA 85,170,170,170,2
      34,186,174,171
IC 560 DATA 85,170,170,170,1
      70,171,174,250
IN 570 DATA 85,171,174,186,2
      34,170,170,170
KD 580 DATA 106,106,106,106,
      106,106,106,106
LE 590 DATA 171,171,171,171,
      171,171,171,171
JN 600 DATA 250,250,250,250,
      250,250,250,250
KF 610 DATA 170,170,170,170,
      170,170,170,170
BD 620 DATA 106,106,106,106,
      107,110,122,85
IJ 630 DATA 171,174,186,234,
      170,170,170,85
IB 640 DATA 250,174,171,170,
      170,170,170,85
IL 650 DATA 170,170,170,234,
      186,174,171,85
ME 660 RESTORE 680:C=0:FOR A
      =29184 TO 30512:READ
      B:POKE A,B:C=C+B:NEXT
       A:IF C=93195 THEN RE
      TURN
KB 670 PRINT "{CLEAR}ERROR I
      N DATA":STOP
HN 680 DATA 160,87,162,114,1
      69,7,32,92,228,104,96
      ,169,10,141,107,116,1
      41,108,116,160,63,169
      ,0,153,47,119
GJ 690 DATA 136,16,250,169,5
      ,141,103,116,133,205,
      169,55,141,98,116,141
      ,100,116,169,3,133,29
      ,133,30,169,0
DB 700 DATA 133,207,133,208,
      133,203,169,7,141,109
      ,116,141,110,116,165,
      206,133,204,162,3,160
      ,0,152,145,203,200
PB 710 DATA 208,251,230,204,
      202,16,246,104,96,216
      ,169,0,133,77,32,126,
      115,206,107,116,208,9
      ,173,108,116,141
BK 720 DATA 107,116,32,237,1
      15,165,28,208,20,206,
      109,116,208,9,173,110
      ,116,141,109,116,32,1
      36,114,32,250,114
JK 730 DATA 32,65,116,76,98,
      228,166,205,165,207,2
      4,125,171,114,201,16,
      144,4,201,241,144,2,1
      33,207,165,208
CN 740 DATA 24,125,179,114,2
      01,16,144,4,201,241,1
      44,2,133,208,96,1,0,2
      55,255,255,0,1,1,255,
      255,255
HO 750 DATA 0,1,1,1,0,206,10
      4,116,208,57,169,19,1
      41,104,116,206,103,11
      6,208,21,169,2,141,10
      3,116,165
AH 760 DATA 29,201,3,240,2,1
      98,29,165,30,201,3,24
      0,2,198,30,165,207,24
      0,9,16,5,230,207,76,2
      36,114
PI 770 DATA 198,207,165,208,
      240,9,16,5,230,208,76
      ,249,114,198,208,96,3
      2,187,114,165,207,141
      ,105,116,166,29
NN 780 DATA 24,101,207,141,1
      05,116,202,208,247,16
      5,208,141,106,116,166
      ,30,24,101,208,141,10
      6,116,202,208,247,173
CN 790 DATA 105,116,48,18,24
      ,109,97,116,141,97,11
      6,173,98,116,105,0,14
      1,98,116,76,78,115,73
      ,255,24,105
KC 800 DATA 1,141,101,116,17
      3,97,116,56,237,101,1
      16,141,97,116,173,98,
      116,233,0,141,98,116,
      173,106,116,48
FB 810 DATA 16,24,109,99,116
      ,141,99,116,173,100,1
      16,105,0,141,100,116,
      96,73,255,24,105,1,14
      1,101,116,173
GA 820 DATA 99,116,56,237,10
      1,116,141,99,116,173,
      100,116,233,0,141,100
      ,116,96,173,98,116,14
      1,0,208,141,1
LJ 830 DATA 208,24,105,8,141
      ,2,208,141,3,208,165,
      205,133,203,169,0,133
      ,204,162,6,6,203,38,2
      04,202,208
PB 840 DATA 249,165,203,24,1
      05,111,141,198,115,16
      5,204,105,116,141,199
      ,115,165,206,133,204,
      169,4,141,102,116,169
BB 850 DATA 0,133,203,172,10
      0,116,145,203,200,162
      ,0,189,255,255,145,20
      3,200,232,224,16,208,
      245,169,0,145,203
JP 860 DATA 173,198,115,24,1
      05,16,141,198,115,173
      ,199,115,105,0,141,19
      9,115,230,204,206,102
      ,116,208,205,96,165
DI 870 DATA 205,201,8,144,1,
      96,173,120,2,74,74,74
      ,176,12,230,205,166,2
      05,224,8,208,4,162,0,
      134,205
II 880 DATA 74,176,8,198,205
      ,16,4,162,7,134,205,1
      73,132,2,208,40,166,2
      05,189,171,114,16,5,7
      3,255,24
JN 890 DATA 105,1,24,101,29,
      201,8,176,2,133,29,18
      9,179,114,16,5,73,255
      ,24,105,1,24,101,30,2
      01,8
HG 900 DATA 176,2,133,30,96,
      165,209,208,27,162,0,
      173,4,208,13,5,208,13
      ,6,208,13,7,208,74,14
      4,2
KI 910 DATA 162,1,74,74,144,
      2,162,2,134,209,96,0,
      0,0,0,0,0,0,0,0,0,0,0
      ,0,0,0
AJ 920 DATA 0,0,0,0,0,0,0,1,3,
      15,19,35,54,28,0,0,0,
      0,0,0,0,0,0,0,0,48,112,
      108
ME 930 DATA 28,9,3,3,0,0,0,8
      ,28,56,112,224,192,12
      8,0,0,0,0,0,0,0,0,16,
      48,0,6
NC 940 DATA 4,0,0,0,0,0,0,0,12
      8,128,0,0,1,1,1,1,1,1
      ,1,1,1,1,1,3,2,2,3
NI 950 DATA 1,4,6,4,0,0,0,0,
      0,0,0,0,4,5,5,4,0,128
      ,128,128,128,128,128,
      128,128,128
IK 960 DATA 128,128,192,64,6
      4,192,128,32,96,32,0,
      0,0,0,0,0,0,0,32,160,
      160,32,0,0,0,16
EL 970 DATA 56,28,14,7,3,1,0
      ,0,0,0,0,0,0,0,8,12,0
      ,96,32,0,0,0,0,0,0,1
JL 980 DATA 1,0,0,0,0,0,0,0,
      0,0,128,192,240,200,1
      96,108,56,0,0,0,0,0,0
      ,0,0,0
ED 990 DATA 0,12,14,54,56,14
      4,192,192,0,0,0,0,0,0
      ,0,0,127,127,127,0,0,
      0,0,0,0,0
ME 1000 DATA 0,0,0,0,112,32,
      0,0,0,32,112,0,0,0,0
      ,0,0,0,0,0,0,60,230,
      230,230,60
JG 1010 DATA 0,0,0,0,0,0,0,0,0
      ,0,56,56,0,24,24,24,
      0,56,56,0,0,0,0,0,0,
      0,0
KG 1020 DATA 0,0,1,3,7,14,28
      ,56,16,0,0,0,0,1,1,0
      ,0,0,0,0,0,0,32,96,0,1
      2,8
NK 1030 DATA 0,0,0,56,108,19
      6,200,240,192,128,0,
      0,0,0,0,0,0,0,192,19
      2,144,56,54,14,12,0
CK 1040 DATA 0,0,0,0,0,0,0,0,1
      ,3,2,2,3,1,1,1,1,1,1
      ,1,1,1,1,1,0,4,5
DC 1050 DATA 5,4,0,0,0,0,0,0,0
      ,0,0,4,6,4,128,192,6
      4,64,192,128,128,128
      ,128,128,128,128,128
EK 1060 DATA 128,128,128,0,3
      2,160,160,32,0,0,0,0
      ,0,0,0,0,0,32,96,32,0,
      0,28,54,35,19,15
BE 1070 DATA 3,1,0,0,0,0,0,0
      ,0,0,3,3,9,28,108,11
      2,48,0,0,0,0,0,0,0,0
      ,0
AJ 1080 DATA 0,0,0,0,0,0,0,128
      ,192,224,112,56,28,8
      ,0,0,0,0,128,128,0,0
      ,0,0,0,0,4
FL 1090 DATA 6,0,48,16,0,0,0
      ,0,0,0,0,0,60,103,103,
      103,60,0,0,0,0,0,0,0,0
      ,0,0,28
JJ 1100 DATA 28,0,24,24,24,0
      ,28,28,0,0,0,0,0,0,0
      ,0,0,0,254,254,254,0
      ,0,0,0,0
LP 1110 DATA 0,0,0,0,0,0,0,14,
      4,0,0,0,4,14,0,0,0,0
      ,0,1,5,0,3,8,11,0,1
JG 1120 DATA 4,10,5,1,4,1,0,
      0,0,0,6,0,0,0,0,6,0,
      0,0,6,0,0,0,0,0,64
EL 1130 DATA 128,64,160,64,1
      60,160,64,128,32,64,
      128,128,0,0,0,0,0,12
      8,0,0,0,0,128,0,192,
      0
BC 1140 DATA 0,0,0,2,0,4,17,
      0,32,10,0,20,64,2,16
      ,1,4,0,0,0,0,0,8,0,0
      ,0
```

```
EE 1150  DATA 0,0,0,0,8,0,0,0
         ,0,0,128,64,0,16,160
         ,8,160,48,0,136,32,0
         ,16,0,128,0
CL 1160  DATA 0,4,0,0,0,0,0,0
         ,0,0,0,64,0,0,0,2,0,
         0,16,0,32,0,0,16,64,
         0
CG 1170  DATA 0,1,0,32,0,0,0,
         0,8,0,0,0,0,0,0,0,16
         ,0,0,0,0,0,128,0,0,1
         6
ME 1180  DATA 0,4,0,0,2,8,0,0
         ,16,0,128,0,0,0,0,0,
         0,0,0,0,0,0,0,0,0,4,0
MN 1190  DATA 0,0,0
```



Watch out for those slick turns in the Apple version of "Miami Ice."

Program 4: Miami Ice For Apple

Version by Tim Victor, Editorial Programmer

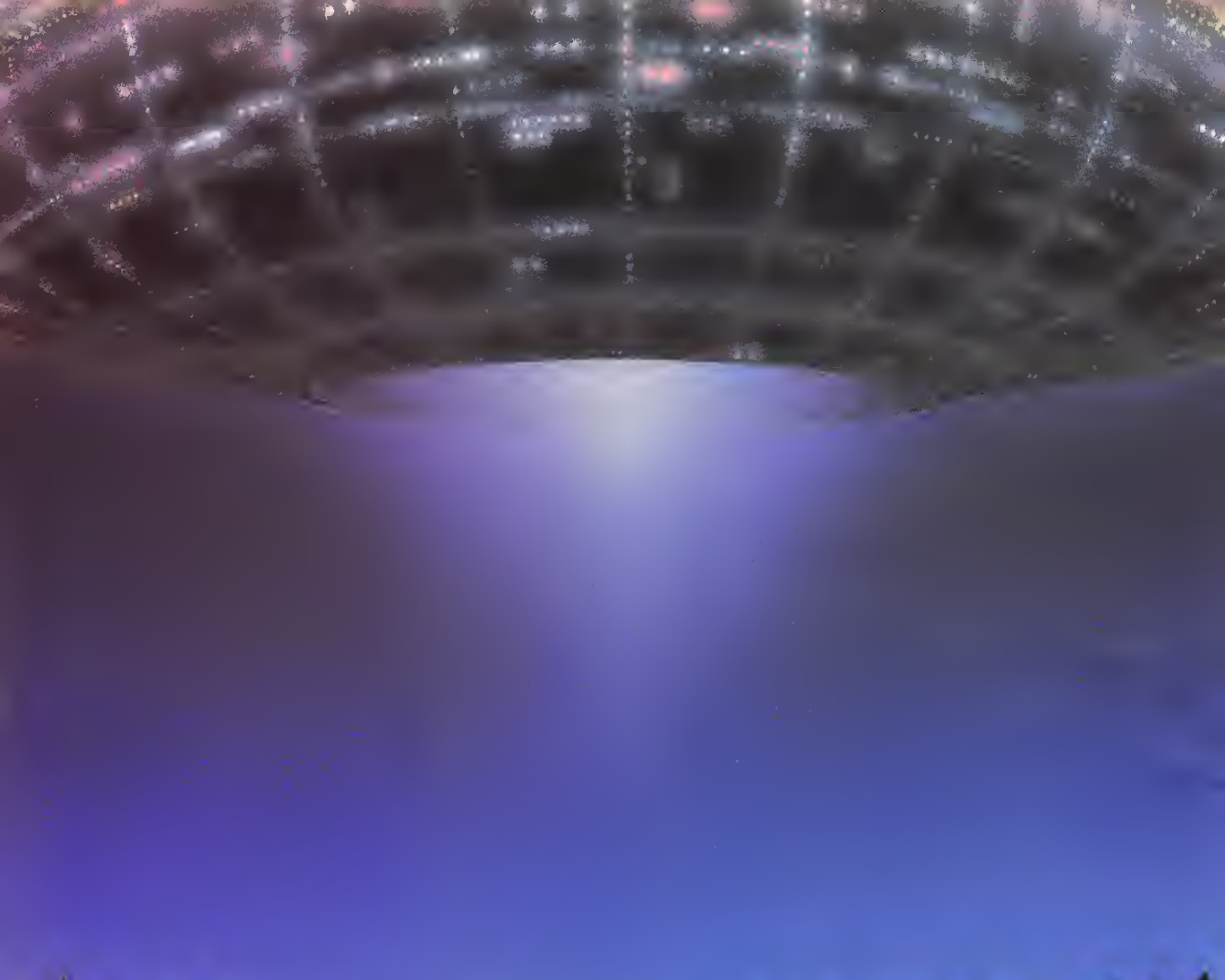
Please refer to the "MLX" article in this issue before entering the following listing.

```
START ADDRESS:  1000
END ADDRESS:    1E97
```

```
1000: A9 00 85 EC A9 60 85 ED 3C
1008: A9 1A 85 FA A9 1A 85 F8 A0
1010: 20 83 17 20 78 17 A9 FF 85
1018: 8D C4 1E 20 97 18 A9 03 69
1020: 8D 88 1E A9 00 8D 8F 1E 28
1028: 8D C0 1E F0 03 20 7E 19 C0
1030: 2C 57 C0 2C 52 C0 2C 54 59
1038: C0 2C 50 C0 20 82 14 A9 77
1040: 00 A0 0A 91 FA A0 11 91 F0
1048: FA 2C 54 C0 A9 20 85 E6 47
1050: A9 40 8D 97 1E A9 01 8D A7
1058: 98 1E A9 01 8D 1D 1A A9 50
1060: 01 8D 1A 1A A9 01 8D 18 D0
1068: 1A A9 0A 8D 1C 1A A9 02 88
1070: 8D 1E 1A A9 00 8D 81 1E 74
1078: A9 00 8D AF 1E 8D B0 1E C0
1080: 20 DA 12 AD 61 C0 30 F8 EE
1088: 20 90 18 EE 89 1E D0 26 DC
1090: C9 7B 90 15 C9 89 90 1E CA
1098: EE 1E 1A AD 1E 1A C9 08 CA
10A0: D0 11 A9 00 8D 1E 1A F0 AC
10A8: 0A CE 1E 1A 10 05 A9 07 D5
10B0: 8D 1E 1A 20 DA 12 AD 61 40
10B8: C0 10 CD 8D A5 1E AE 1E F0
10C0: 1A 8D E3 12 8D A6 1E 8D F8
10C8: EB 12 8D A7 1E A9 01 8D 86
10D0: A8 1E A9 00 8D 82 1E A9 1F
10D8: 00 8D C1 1E 8D C2 1E 20 4A
10E0: 90 18 C9 78 90 08 E9 10 98
10E8: C9 79 80 02 A9 78 18 69 48
10F0: 08 4A 4A 4A 4A 38 E9 08 A4
10F8: 8D A9 1E 48 AD A8 1E 69 48
1100: F8 2C A9 1E 30 0C 49 FF 05
1108: 38 69 00 CD A9 1E B0 0A AE
1110: 90 05 CD A9 1E 90 03 8D D6
1118: A9 1E 68 F0 19 38 ED A9 E1
1120: 1E F0 0A 30 04 A9 02 D0 6D
1128: 0D A9 FE D0 09 A9 01 2C 45
1130: A9 1E 10 02 A9 FF 8D AA E3
1138: 1E AD 61 C0 10 0F 2C A5 C7
1140: 1E 30 0A A0 08 CC A8 1E A8
1148: 90 03 EE A8 1E 8D A5 1E 6C
1150: AD A9 1E AC A8 1E 20 FD 3E
1158: 12 A5 51 48 AC A6 1E 20 F7
1160: F3 12 AD A7 1E 18 65 50 9D
1168: 8D A7 1E 68 AC A7 1E 20 E5
1170: F3 12 AD A6 1E 38 E5 50 1F
1178: 8D A6 1E AD A8 1E AC A6 67
1180: 1E 20 FD 12 A5 50 8D A8 CF
1188: 1E A5 51 8D AC 1E AD A8 08
1190: 1E AC A7 1E 20 FD 12 A5 86
1198: 50 8D AD 1E A5 51 8D AE 1A
11A0: 1E AC 82 1E C8 D0 0C A0 77
11A8: 00 AD A8 1E C9 01 F0 03 64
11B0: CE A8 1E 8C 82 1E EE C1 9E
11B8: 1E D0 0A EE C2 1E D0 05 83
11C0: A9 FF 8D C2 1E 18 AD AF F1
11C8: 1E 6D AC 1E 8D AF 1E AD E1
11D0: 18 1A 6D A8 1E C9 07 2C C1
11D8: A8 1E 30 09 90 0E EE 1A A3
11E0: 1A E9 07 80 07 90 05 CE C9
11E8: 1A 1A 69 06 18 6D 81 1E 24
11F0: C9 07 90 05 E9 07 EE 1A 7F
11F8: 1A 8D 18 1A 4D 1A 1A 29 C0
1200: 01 8D 81 1E F0 11 CE 18 A4
1208: 1A 10 0C AD 18 1A 18 69 74
1210: 07 8D 18 1A CE 1A 1A 18 48
1218: AD 80 1E 6D AE 1E 8D 80 93
1220: 1E AD 1C 1A 6D AD 1E 8D CF
1228: 1C 1A AD A6 1E C9 20 90 E9
1230: 0A C9 E0 80 06 C9 00 10 5A
1238: 0D 30 21 A0 00 2C A7 1E 38
1240: 30 2E A0 04 D0 2A AD A7 8E
1248: 1E A0 02 C9 E0 80 21 C9 56
1250: 20 90 1D C9 00 10 02 88 85
1258: 24 C8 D0 14 AD A7 1E A0 05
1260: 06 C9 20 90 08 C9 E0 80 F8
1268: 07 C9 00 10 02 C8 24 88 87
1270: 98 18 6D AA 1E 10 03 18 8E
1278: 69 08 C9 08 90 02 E9 08 75
1280: 8D 1E 1A 20 DA 12 AD 88 68
1288: 1E F0 37 AD 8A 1E F0 35 1F
1290: A9 20 38 ED C2 1E 90 1E 45
1298: 0A 0A 8D C5 1E AE 83 1E 83
12A0: AD C5 1E 18 6D 8F 1E C9 C2
12A8: 64 90 05 EE C0 1E E9 64 69
12B0: 8D 8F 1E CA 10 EA AD 83 37
12B8: 1E C9 06 F0 1A EE 83 1E 40
12C0: D0 15 4C DF 10 A2 00 A0 C5
12C8: 00 C8 D0 FD E8 D0 F8 CE 64
12D0: 88 1E D0 03 4C 18 10 4C DF
12D8: 2D 10 20 2F 14 20 2F 13 21
12E0: 4C F2 16 00 2D 40 2D 00 6F
12E8: D3 C0 D3 C8 D3 00 2D 40 E6
12F0: 2D 00 D3 20 FD 12 24 51 F9
12F8: 10 02 E6 50 60 85 4E 84 C1
1300: 4F A9 00 85 50 24 4E 10 50
1308: 05 38 A9 00 E5 4F 85 51 8C
1310: A2 08 06 51 26 50 06 4E 2C
1318: 90 11 18 A5 51 65 4F 85 6C
1320: 51 90 02 E6 50 24 4F 10 83
1328: 02 C6 50 CA D0 E4 60 A0 33
1330: 03 81 FA D0 01 60 A0 04 7F
1338: 81 FA 8D A2 1E 0A 18 69 84
1340: 19 A8 81 FA 85 FC C8 81 66
1348: FA 85 FD A0 00 81 FA 18 E8
1350: 71 FC 8D 9D 1E C8 81 FA 6C
1358: 18 71 FC C9 07 90 05 EE 96
1360: 9D 1E E9 07 8D 9E 1E C8 76
1368: 81 FA 18 71 FC 8D 9F 1E 88
1370: AD 9E 1E 0A 0A 18 69 05 02
1378: A8 81 FC 85 1C C8 81 FC 88
1380: 85 1D C8 81 FC 85 1E C8 E7
1388: 81 FC 85 1F A9 0A AC 98 D0
1390: 1E F0 03 18 69 07 A8 A9 46
1398: 01 91 FA C8 AD 9D 1E 91 41
13A0: FA C8 AD 9E 1E 91 FA C8 0C
13A8: AD 9F 1E 91 FA C8 AD A2 63
13B0: 1E 91 FA C8 81 FA 85 EE A9
13B8: CB B1 FA 85 EF A0 03 81 21
13C0: FC 8D A0 1E C8 81 FC 8D 53
13C8: A1 1E A9 00 8D 88 1E 8D 95
13D0: 8A 1E AD 9F 1E 8D A3 1E 18
13D8: AD A1 1E 8D A4 1E 20 0F C7
13E0: 17 AC A0 1E 88 81 FE 91 4E
13E8: EE 31 1E D1 1E F0 0A EE 68
13F0: 88 1E C9 00 30 03 EE 8A 5A
13F8: 1E 51 FE 11 1C 91 FE 88 21
1400: 10 E3 18 A5 EE 6D A0 1E 13
1408: 85 EE 90 02 E6 EF 18 A5 AD
1410: 1C 6D A0 1E 85 1C 90 02 57
1418: E6 1D 18 A5 1E 6D A0 1E 5E
1420: 85 1E 90 02 E6 1F EE A3 F9
1428: 1E CE A4 1E D0 80 60 A9 3D
1430: 0A AC 98 1E F0 03 18 69 AA
1438: 07 A8 81 FA D0 01 60 A9 E8
1440: 00 91 FA C8 81 FA 8D 9D EA
1448: 1E C8 81 FA 8D 9E 1E C8 83
1450: 81 FA 8D 9F 1E C8 81 FA 2E
1458: 8D A2 1E C8 81 FA 85 EE 83
1460: C8 81 FA 85 EF AD A2 1E AA
1468: 0A 18 69 19 A8 81 FA 85 E1
1470: FC C8 81 FA 85 FD A0 03 97
1478: 81 FC 8D A8 1E C8 81 FC E8
1480: 8D A1 1E AD 9F 1E 8D A3 AA
1488: 1E AD A1 1E 8D A4 1E 20 9C
1490: 0F 17 AC A0 1E 88 81 EE 08
1498: 91 FE 88 10 F9 18 A5 EE C5
14A0: 6D A0 1E 85 EE 90 02 E6 68
14A8: EF EE A3 1E CE A4 1E D0 F0
14B0: DE 60 A9 FF 85 1C A9 20 A5
14B8: 85 E6 8D 97 1E 20 F6 F3 D8
14C0: A9 00 8D A3 1E 8D 9D 1E 2A
14C8: 20 0F 17 A9 80 A0 27 91 A8
14D0: FE 88 10 F8 EE A3 1E 20 8E
14D8: 0F 17 A9 80 A0 00 91 FE 82
14E0: A0 27 91 FE EE A3 1E AD 35
14E8: A3 1E C9 88 D0 E9 20 0F AC
14F0: 17 A9 80 A0 27 91 FE 88 2F
14F8: 10 F8 EE A3 1E AD A3 1E 4D
1500: C9 C0 D0 EA A9 00 8D 9D 0E
1508: 1E A9 F8 8D 9F 1E AD 83 28
1510: 1E 0A AA 8D A8 16 85 1C C1
1518: 8D A9 16 85 1D A0 00 8C 9E
1520: 87 1E AC 87 1E 81 1C F0 9F
1528: 66 EE 87 1E 48 8D 84 1E 1A
1530: 29 1F 8D 85 1E 68 29 60 09
1538: 8D 86 1E AD 86 1E F0 12 97
1540: C9 40 F0 1A 90 06 EE 9D 37
1548: 1E 4C 67 15 CE 9D 1E 4C 48
1550: 67 15 38 AD 9F 1E E9 08 A6
1558: 8D 9F 1E 4C 67 15 18 AD 27
1560: 9F 1E 69 08 8D 9F 1E 2C E2
1568: 84 1E 10 18 AD 9F 1E 8D DD
1570: A3 1E A9 80 A2 00 20 0F 95
1578: 17 A9 80 81 FE EE A3 1E D9
1580: A9 07 2C A3 1E D0 EF CE E3
1588: 85 1E D0 AF 4C 22 15 AC E8
1590: 87 1E C8 81 1C 8D 9D 1E C2
1598: C8 81 1C 8D A3 1E A9 86 8F
15A0: 85 1C A9 16 85 1D A9 14 33
15A8: 8D A4 1E 20 0F 17 A0 02 A0
15B0: 81 1C 91 FE 88 10 F9 EE 44
15B8: A3 1E 18 A5 1C 69 03 85 A8
15C0: 1C 90 02 E6 1D CE A4 1E 57
15C8: D0 E1 A9 00 85 1C 85 FE AF
15D0: A9 20 85 1D A9 40 85 FF 83
15D8: A0 00 81 1C 91 FE C8 D0 36
15E0: F9 E6 1D E6 FF A5 FF C9 34
15E8: 60 D0 EF 60 48 FF 86 CD 80
15F0: 00 07 7E 6A D1 A3 E6 66 A4
15F8: C6 90 A3 E6 65 86 C5 E3 42
1600: CC A3 E6 00 1D 0E 45 EC F0
1608: 47 25 EE 88 6A 43 CD AA 5E
1610: C2 02 84 45 65 00 05 96 D4
1618: 68 C3 45 A7 E8 C3 83 E4 E6
1620: 83 C3 E6 C3 83 E2 A1 04 07
1628: 83 47 E6 83 C3 E3 C3 45 77
1630: E6 AC 83 C6 83 AA 83 C6 6C
1638: 83 AA 83 C6 45 65 83 00 74
1640: 1D 88 66 C6 A1 E3 A2 C8 01
1648: E1 27 E1 A1 66 C4 45 24 7A
1650: E1 63 05 EA C2 82 EA 81 0D
1658: C1 E5 46 81 05 E3 06 E4 68
1660: A1 C2 82 AD 81 C1 A6 C2 5C
1668: 82 A4 84 E1 A1 82 A1 E1 E9
1670: 81 68 05 C3 A1 E1 C3 EA 58
```

```
1678: C1 81 E2 06 C1 00 19 0C EE
1680: 65 D3 E9 64 E2 81 C1 EE 67
1688: 8E A2 24 AA 24 A4 C9 E8 04
1690: 64 EC 00 0D 3F 46 F2 82 76
1698: E1 C2 EC C5 E2 C1 A2 C4 88
16A0: AC C2 A1 82 81 00 05 94 5C
16A8: EC 15 95 16 F3 15 80 16 AF
16B0: 06 16 42 16 18 16 D5 AA 7E
16B8: D5 D5 AA D5 D5 AA 00 D5 27
16C0: AA 00 D5 AA 00 D5 AA 00 54
16C8: D5 AA 00 D5 AA 00 D5 AA 93
16D0: 00 D5 AA 00 D5 AA 00 D5 F6
16D8: AA 00 D5 AA 00 D5 AA 00 6C
16E0: D5 AA 00 D5 AA 00 D5 AA A8
16E8: 00 D5 AA 00 D5 AA D5 D5 8A
16F0: AA D5 AD 97 1E C9 40 A9 59
16F8: 00 2A AA 8D 54 C0 8A 49 E4
1700: 01 8D 98 1E AD 97 1E 85 94
1708: E6 49 60 8D 97 1E 60 AD 84
1710: A3 1E 29 3F A8 89 38 17 64
1718: 0D 97 1E 85 FF AD A3 1E EA
1720: 29 08 F0 02 A9 80 18 2C CE
1728: A3 1E 70 04 10 04 69 28 89
1730: 69 28 6D 9D 1E 85 FE 60 0A
1738: 00 04 08 0C 10 14 18 1C 46
1740: 00 04 08 0C 10 14 18 1C 4E
1748: 01 05 09 0D 11 15 19 1D 56
1750: 01 05 09 0D 11 15 19 1D 5E
1758: 02 06 0A 0E 12 16 1A 1E 66
1760: 02 06 0A 0E 12 16 1A 1E 6E
1768: 03 07 08 0F 13 17 18 1F 76
1770: 03 07 08 0F 13 17 18 1F 7E
1778: A9 00 A0 0A 91 FA A0 11 FA
1780: 91 FA A0 0F A5 EC 91 FA 3A
1788: C8 A5 ED 91 FA A0 09 81 79
1790: FA 18 65 EC 85 EC 90 02 C0
1798: E6 ED A0 16 A5 EC 91 FA 2A
17A0: C8 A5 ED 91 FA A0 09 81 91
17A8: FA 18 65 EC 85 EC 90 02 D8
17B0: E6 ED 60 A9 00 8D A2 1E 0E
17B8: AD A2 1E 0A 69 19 A8 81 7D
17C0: FA 85 FC C8 81 FA 85 FD 7C
17C8: 20 D8 17 EE A2 1E AD A2 9A
17D0: 1E A0 18 D1 FA D0 E1 60 95
17D8: A9 01 8D 9E 1E A0 03 81 E2
17E0: FC 8D A0 1E C8 81 FC 8D 78
17E8: A1 1E A9 09 8D 99 1E A0 E4
17F0: 05 81 FC 85 1C C8 81 FC 6A
17F8: 85 1D AC 99 1E A5 EC 91 53
1800: FC C8 A5 ED 91 FC C8 8C 13
1808: 99 1E 20 36 18 A0 07 81 F6
1810: FC 85 1C C8 81 FC 85 1D D9
1818: AC 99 1E A5 EC 91 FC C8 93
1820: A5 ED 91 FC C8 8C 99 1E 6A
1828: 20 36 18 EE 9E 1E AD 9E 4F
1830: 1E C9 07 D0 8A 60 AD A1 24
1838: 1E 8D A4 1E A9 00 8D 9A 54
1840: 1E A0 00 A9 00 8D 98 1E CD
1848: 81 1C 8D 9C 1E 0A AE 9E E8
1850: 1E 0A 2E 98 1E CA D0 F9 49
1858: 2C 9C 1E 10 02 38 24 18 D8
1860: 6A 0D 9A 1E 91 EC AD 98 75
1868: 1E 8D 9A 1E C8 CC A0 1E 19
1870: D0 D1 18 A5 1C 6D A0 1E D0
1878: 85 1C 90 02 E6 1D 18 A5 26
1880: EC 6D A0 1E 85 EC 90 02 78
1888: E6 ED CE A4 1E D0 AD 60 88
1890: A2 00 20 1E F8 98 60 2C 27
1898: 54 C0 2C 51 C0 20 58 FC F1
18A0: 2C C4 1E 30 3E A9 04 85 05
18A8: 25 20 22 FC A9 0F 85 24 40
18B0: A2 00 8D 0E 1A F0 06 20 88
18B8: ED FD E8 D0 F5 AD 8F 1E 8D
18C0: C9 0A 90 07 EE 17 06 E9 A4
18C8: 0A 80 F5 69 80 8D 18 06 71
18D0: AD C0 1E C9 0A 90 07 EE F7
18D8: 15 06 E9 0A 80 F5 69 80 D3
18E0: 8D 16 06 A9 00 8D C4 1E 96
18E8: 20 2A 19 A9 00 8D 83 1E 2D
18F0: AD 61 C0 30 05 8D A5 1E 33
18F8: 10 1D 2C A5 1E 30 18 8D C7
1900: A5 1E EE B3 1E AD B3 1E D2
1908: C9 07 D0 05 A9 00 8D B3 67
1910: 1E 18 69 B1 8D 8F 04 A0 83
1918: 00 A2 00 CA D0 FD 88 D0 FF
1920: FA AD 00 C0 10 CA 2C 10 58
1928: C0 60 A2 00 8D 4E 19 F0 71
1930: 1C 85 25 20 22 FC E8 8D 0D
1938: 4E 19 85 24 E8 8D 4E 19 8E
1940: F0 07 E8 20 ED FD 4C 3D 09
1948: 19 E8 4C 2C 19 60 07 10 F5
1950: CD C9 C1 CD C9 A0 C9 C3 19
1958: C5 00 09 11 CC C5 D6 C5 90
1960: CC A0 81 00 0C 09 D0 D2 50
1968: C5 D3 D3 A0 C1 CE D9 A0 94
1970: C8 C5 D9 A0 D4 CF A0 C2 29
1978: C5 C7 C9 CE 00 00 2C 54 52
1980: C0 2C 51 C0 20 58 FC A2 53
1988: 00 8D FD 19 F0 1A 85 25 98
1990: 20 22 FC E8 8D FD 19 85 27
1998: 24 E8 8D FD 19 F0 06 20 67
19A0: ED FD E8 D0 F5 E8 D0 E1 4A
19A8: AD 83 1E 18 69 81 8D 96 A7
19B0: 07 18 AD 88 1E 69 80 8D 63
19B8: 8E 04 AD 8F 1E C9 0A 90 89
19C0: 07 EE 8F 05 E9 0A 80 F5 49
19C8: 69 80 8D C0 05 AD C0 1E 18
19D0: C9 0A 90 07 EE 8D 05 E9 4F
19D8: 0A 80 F5 69 80 8D 8E 05 CF
19E0: A9 02 8D 8C 1E A9 00 8D 08
19E8: 8D 1E 8D 8E 1E CE 8E 1E E6
19F0: D0 F8 CE 8D 1E D0 F6 CE 31
19F8: 8C 1E D0 F1 60 07 10 CC 56
1A00: C5 D6 C5 CC 00 09 11 C3 5C
1A08: C1 D2 D3 00 08 0F D3 C3 4C
1A10: CF D2 C5 A0 80 80 80 80 FD
1A18: 00 00 20 20 20 53 20 20 01
1A20: 32 45 20 40 20 20 20 20 AB
1A28: 20 00 20 20 20 20 20 7F 83
1A30: 20 20 08 43 1A 64 1A 85 CD
1A38: 1A A6 1A C7 1A E8 1A 09 94
1A40: 18 2A 18 01 03 00 03 11 2F
1A48: 48 18 7E 18 20 35 20 20 A0
1A50: 20 20 20 20 20 20 20 46 AA
1A58: 2E 44 20 20 20 20 41 20 DE
1A60: 20 20 20 20 01 01 01 04 C4
1A68: 10 97 1C D7 1C 20 20 20 4D
1A70: 20 20 20 20 20 20 20 20 A4
1A78: 20 20 3A 20 20 20 20 20 EF
1A80: 20 4F 20 0F 23 01 04 08 8A
1A88: 04 08 17 1C 37 1C 20 30 FF
1A90: 00 20 30 20 20 36 20 20 0F
1A98: 20 20 49 20 20 45 20 20 86
1AA0: 20 20 20 20 20 20 01 01 77
1AA8: 06 04 10 17 1D 57 1D 34 09
1AB0: 20 31 20 20 20 2E 20 53 94
1AB8: 45 20 20 20 20 20 20 20 7F
1AC0: 20 20 20 20 20 20 00 01 95
1AC8: 03 06 03 11 81 18 E4 18 50
1AD0: 20 20 20 20 4F 20 55 20 E8
1AD8: 20 20 20 20 20 20 31 2E 3D
1AE0: 30 31 20 20 00 46 00 20 88
1AE8: 00 06 06 04 10 17 1E 57 10
1AF0: 1E 00 20 20 20 20 38 20 52
1AF8: 20 20 20 2E 7F 20 20 20 09
1B00: 24 20 20 20 7F 20 20 20 33
1B08: 20 00 06 08 04 08 57 1C 9A
1B10: 77 1C 20 20 20 41 20 44 99
1B18: 20 20 20 20 20 20 20 52 B0
1B20: 20 20 7F 20 20 4F 20 20 FE
1B28: 20 20 00 06 01 04 10 97 A6
1B30: 1D D7 1D 7F 20 22 20 20 70
1B38: 20 20 20 20 20 20 20 20 6E
1B40: 20 20 20 20 46 20 20 20 A7
1B48: 49 20 20 D2 84 80 D5 8A 88
1B50: 80 D5 8A 80 D5 8A 80 D5 45
1B58: 8A 80 D5 8A 80 D5 8A 80 48
1B60: FD 88 80 9F 8F 80 81 88 8C
1B68: 80 81 88 80 81 88 80 85 0D
1B70: 8A 80 D5 8A 80 D5 8A 80 60
1B78: D5 8A 80 89 89 80 FE 87 88
1B80: 80 FF 8F 80 FF 8F 80 FF 30
1B88: 8F 80 FF 8F 80 FF 8F 80 43
1B90: FF 8F 80 FF 8F 80 FF 8F C8
1B98: 80 FF 8F 80 FF 8F 80 FF 48
1BA0: 8F 80 FF 8F 80 FF 8F 80 58
1BA8: FF 8F 80 FF 8F 80 FF 8F E0
1BB0: 80 89 89 80 D5 8A 80 D5 72
1BB8: 8A 80 D5 8A 80 85 8A 80 67
1BC0: 81 88 80 81 88 80 81 88 D3
1BC8: 80 9F 8F 80 FD 88 80 D5 16
1BD0: 8A 80 D5 8A 80 D5 8A 80 C0
1BD8: D5 8A 80 D5 8A 80 D5 8A 96
1BE0: 80 D3 84 80 FF 8F 80 FF 24
18E8: 8F 80 FF 8F 80 FF 8F 80 A3
18F0: FF 8F 80 FF 8F 80 FF 8F 29
18F8: 80 FF 8F 80 FF 8F 80 FF A8
1C00: 8F 80 FF 8F 80 FF 8F 80 8C
1C08: FF 8F 80 FF 8F 80 FF 8F 42
1C10: 80 FF 8F 80 FE 87 80 A9 43
1C18: DD EA 80 AD F1 8A 81 AD 4F
1C20: E0 AA 81 A9 E0 AA 81 A9 9C
1C28: E0 AA 81 AD E0 AA 81 AD E8
1C30: F1 8A 81 A9 DD EA 80 FF 6A
1C38: FF FF 80 FF FF FF 81 FF 83
1C40: FF FF 81 FF FF FF 81 FF A8
1C48: FF FF 81 FF FF FF 81 FF 83
1C50: FF FF 81 FF FF FF 80 D6 90
1C58: 8A 95 81 D1 8E 85 81 D5 C4
1C60: 86 84 81 D5 86 94 81 D5 F5
1C68: 86 94 81 D5 86 84 81 D1 72
1C70: 8E 85 81 D6 8A 95 81 FE 29
1C78: FF FF 81 FF FF FF 81 FF E3
1C80: FF FF 81 FF FF FF 81 FF E8
1C88: FF FF 81 FF FF FF 81 FF F3
1C90: FF FF 81 FE FF FF 81 00 E8
1C98: A0 85 80 00 A8 94 80 00 28
1CA0: A8 95 80 00 AA D5 80 00 50
1CA8: AA C5 81 C0 8E D5 80 C0 F2
1CB0: 80 95 80 D0 E0 95 80 90 82
1CB8: C0 85 80 94 C0 85 80 D2 F8
1CC0: 80 81 00 CA A0 81 00 D4 26
1CC8: AA 80 00 C8 AA 80 00 90 EA
1CD0: 8A 80 00 A0 8D 80 00 00 E6
1CD8: E0 87 80 00 F8 9F 80 00 8A
1CE0: F8 9F 80 00 FE FF 80 00 86
1CE8: FE FF 81 C0 FF FF 80 C0 9E
1CF0: FF 9F 80 F0 FF 9F 80 F0 A0
1CF8: FF 87 80 FC FF 87 80 FE 11
1D00: FF 81 00 FE FF 81 00 FC 8D
1D08: 8F 80 00 F8 8F 80 00 F0 C2
1D10: 8F 80 00 E0 8F 80 00 A0 5F
1D18: 8D 80 00 90 8A 80 00 C8 61
1D20: AA 80 00 D4 AA 80 00 CA 3F
1D28: A0 81 00 D2 80 81 00 94 DE
1D30: C0 85 80 90 C0 85 80 D0 33
1D38: E0 95 80 C0 80 95 80 C0 02
1D40: 8E D5 80 00 AA C5 81 00 CE
1D48: AA D5 80 00 A8 95 80 00 F9
1D50: A8 94 80 00 A0 85 80 E0 11
1D58: 8F 80 00 F0 8F 80 00 F8 01
1D60: 8F 80 00 FC 8F 80 00 FE 69
1D68: FF 81 00 FE FF 81 00 FC F5
1D70: FF 87 80 F0 FF 87 80 F0 88
1D78: FF 9F 80 C0 FF 9F 80 C0 F6
1D80: FF FF 80 00 FE FF 81 00 C5
1D88: FE FF 80 00 F8 9F 80 00 99
1D90: F8 9F 80 00 E0 87 80 D0 36
1D98: B2 80 00 94 8A 80 00 D4 A8
1DA0: 8A 80 00 D5 AA 80 00 D1 C6
1DA8: AA 80 00 D5 8E 81 00 D4 86
1DB0: 86 81 00 D4 83 85 80 D0 DF
1DB8: 81 84 80 D0 81 94 80 C0 12
1DC0: 80 A5 80 C0 82 A9 80 00 7C
1DC8: AA 95 80 00 AA 89 80 00 4A
1DD0: A8 84 80 00 D0 82 80 F0 13
1DD8: B3 80 00 FC 8F 80 00 FC 40
1DE0: 8F 80 00 FF 8F 80 00 FF 03
1DE8: 8F 80 00 FF FF 81 00 FC 26
1DF0: FF 81 00 FC FF 87 80 F0 68
1DF8: FF 87 80 F0 FF 9F 80 C0 74
1E00: FF 8F 80 C0 FF 8F 80 00 48
1E08: FE 9F 80 00 FE 8F 80 00 F2
1E10: F8 87 80 00 F0 83 80 00 51
1E18: D0 82 80 00 A8 84 80 00 C5
1E20: AA 89 80 00 AA 95 80 C0 91
1E28: 82 A9 80 C0 80 A5 80 D0 98
1E30: 81 94 80 D0 81 84 80 D4 63
1E38: 83 85 80 D4 86 81 00 D5 05
1E40: 8E 81 00 D1 AA 80 00 D5 86
1E48: AA 80 00 D4 8A 80 00 94 32
1E50: 8A 80 00 D0 82 80 00 00 15
1E58: F0 83 80 00 F8 87 80 00 E4
1E60: FE 8F 80 00 FE 9F 80 C0 48
1E68: FF 8F 80 C0 FF 8F 80 F0 A1
1E70: FF 9F 80 F0 FF 87 80 FC CF
1E78: FF 87 80 FC FF 81 00 FF 7C
1E80: FF 81 00 FF 8F 80 00 FF 1D
1E88: 8F 80 00 FC 8F 80 00 FC 10
1E90: 8F 80 00 F0 83 80 00 00 E1
```

©



UFO Invasion
For IBM And Amiga

John Robinson

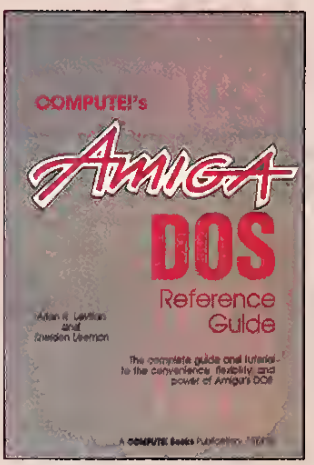
This arcade game sets up a classic confrontation—you, the lone defender, against a horde of deadly machines from outer space. The original IBM version requires Cartridge BASIC for the PCjr or BASICA and a color/graphics adapter for the PC. We've also translated the game for the Amiga computer.

You are at the controls of a surface-to-air missile launcher located somewhere near the Arctic Circle. Your radar outposts have detected a fleet of unidentified flying objects (UFOs) invading the North American continent. Forewarned about this possibility, you know that the waves of UFOs will attack your missile site before

AMIGA

SUPPORT FROM COMPUTE! BOOKS

Everything for the Amiga. From BASIC beginner's guides to advanced programming handbooks, COMPUTE! offers you information-packed tutorials, reference guides, programming examples, ready-to-enter applications, and games to help you develop your computing skills on Commodore's Amiga.



COMPUTE!'s AmigaDOS Reference Guide

Arlan R. Levitan and Sheldon Leemon

A comprehensive tutorial and reference guide to the powerful AmigaDOS—the operating system underlying the Workbench and Intuition—this book offers information useful to every Amiga owner. It defines and illustrates all DOS commands, and shows you how to create file directories, access peripherals, run batch file programs, and avoid "disk shuffle." The screen- and line-oriented text editors are explained in detail. Numerous examples and techniques explain how to use AmigaDOS to make operating your Amiga both convenient and efficient.

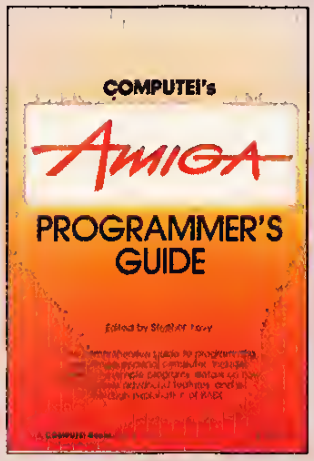
$14.95 ISBN 0-87455-047-5

Elementary Amiga BASIC

C. Regena

Here's your introduction to the new and powerful BASIC on the Amiga personal computer. The Amiga's impressive graphics, animation, and sound can be unlocked with the right commands, and BASIC is the place to start. Complete descriptions of Amiga BASIC's commands, syntax, and organization take you from the beginner level to a full-fledged programmer. Plus, the book offers you ready-to-type-in programs and subroutines while showing you how to write your own programs. *There is a disk available which includes the programs in the book, $15.95. This title is also available as a book/disk combination for $29.95 (057-2)*

$14.95 ISBN 0-87455-041-6



COMPUTE!'s Amiga Programmer's Guide

Edited

Your tutorial and reference manual to AmigaDOS, BASIC, Intuition, and other important software tools which accompany the new Amiga, *COMPUTE!'s Amiga Programmer's Guide* is a clear and thorough guide to the inner workings of this fascinating new-generation computer. The great speed of its 68000 microprocessor, coupled with the versatility of the Amiga-specific graphics and sound, makes the Amiga one of the most powerful computers available today. This book is the key to accessing the Amiga's speed and power.

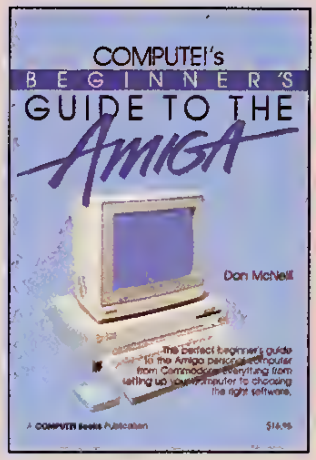
$16.95 ISBN 0-87455-028-9

Advanced Amiga BASIC

Tom R. Halfhill and Charles Brannon

This guide to applications programming on Commodore's new Amiga contains everything an intermediate programmer requires to begin creating sophisticated software on this powerful machine, including several ready-to-type-in programs. Clear, yet comprehensive documentation and examples cover advanced BASIC commands, designing graphic applications, generating sound and music, using the Amiga's built-in speech synthesizer, creating a user interface, and programming the computer's peripherals. *There is a disk available which includes the programs in the book, $15.95.* (June release)

$16.95 ISBN 0-87455-045-9



COMPUTE!'s Beginners Guide to the Amiga

Dan McNeill

Written in a lively and entertaining style, this book teaches you everything a beginner needs to know to get started quickly with the Amiga from Commodore. You will learn about setting up the system, all the most popular types of software, and details about the hardware.

$16.95 ISBN 0-87455-025-4

Inside Amiga Graphics

Sheldon Leemon

The Amiga, Commodore's powerful new computer, is an extraordinarily impressive graphics machine. Easy to use, the Amiga can produce color graphics and excellent animation. You'll find thorough descriptions of the computer's abilities and the hardware required to create a complete graphics system. Software, too, is central to the Amiga's power, and complete tutorials show you how to get the most from the machine. (June release)

$16.95 ISBN 0-87455-040-8

COMPUTE!'s Kids and the Amiga

Edward H. Carlson

The latest in this bestselling series written by Edward Carlson, *COMPUTE!'s Kids and the Amiga*, will acquaint you with BASIC. Over 30 sections—all with instructor notes, lessons, assignments, and lively illustrations—entertain and amuse you as you learn to program your new computer. Clear writing and concise examples make it easy for anyone—children and adults alike—to painlessly learn BASIC. (May release)

$14.95 ISBN 0-87455-048-3

To order, use the enclosed order form.
Or call toll-free 1-800-638-3822 (in NY 212-887-8527).

COMPUTE! Publications, Inc.
Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies
825 7th Avenue, 6th Floor, New York, NY 10019
Publishers of COMPUTE!, COMPUTE!'s Gazette, COMPUTE!'s Gazette Disk, COMPUTE! Books, and COMPUTE!'s Apple Applications

COMPUTE! books are available in the U.K., Europe, the Middle East, and Africa from Holt Saunders, Ltd., 1 St. Anne's Road, Eastbourne, East Sussex BN21 3UN, England, and in Canada from McGraw-Hill, Ryerson Ltd., 330 Progress Ave., Scarborough, Ontario, Canada M1P 2Z5.

proceeding to the more populous areas lying southward. Are you up to the challenge?

Though it's not a particularly long program, "UFO Invasion" offers quite a test for your gaming skills, particularly at the higher levels. Type in the version for your computer and save a copy before you run it. As the screen photos illustrate, both versions look and play almost identically.

Since the Amiga version doesn't use line numbers, we've used a special character (a left arrow) to show you where each program line ends. Don't try to type in the arrows—they're present only to show you where each line ends. (Actually, you can't type an arrow even if you want to, since we deliberately picked a symbol that's not available on the Amiga keyboard.) Instead, wherever you see a left arrow in the listing, you should press RETURN or move the cursor off the line to enter it into memory. To illustrate, look at these program lines:

```
DEFINT A-Z←
RANDOMIZE TIMER←
SCREEN 1, 320, 200, 2, 1←
WINDOW 1,"UFO Invasion",(0,0)-(3
11,185),20,1←
```

The first three lines are short enough to fit into one magazine column, but the fourth program line is so long that it wraps around onto a second line. The arrow shows you that the line ends after the final 1, not after the 3.

First Line Of Defense

When UFO Invasion begins, you'll hear the sound of an alarm siren and see two warning messages scroll across the screen. The middle of the screen contains your control panel. The observatory window at the top gives you a direct view of the skyline in your defense sector. Within the window is the aiming crosshair for your missile launcher, and directly below is a radar screen.

When the saucer-shaped UFO appears, your job is to move the crosshair onto the UFO (using the cursor keys) and launch a missile at it (by pressing the space bar). If your missile hits the UFO, the automated craft is vaporized immediately.

Before it can fire at your base, the UFO must locate your position. Once your position is located, the UFO is certain to hit the mark. Your force shields are powerful enough to protect you against three hits by the UFO, but the fourth hit neutralizes your defenses and paves the way for a successful invasion (ending the game as well).

Control Panel

The control panel is equipped with six gauges to help you monitor events. On each side of the circular radar screen are two ladder gauges. The gauge at the far right shows you how many UFOs remain to be eliminated in the current level. There are eight levels in all; you must eliminate 29 UFOs at each level before advancing to the next.

The gauge directly to the right of the radar screen indicates how close the UFO is to locating your position. When this indicator reaches the top, the UFO scores a hit.

The gauge at the far left shows your points for the current level. You receive 100 points (shown as one bar on the ladder) for each UFO you destroy, with an additional bar for hitting the UFO before the timer is halfway to the top. If you score two bars for every UFO on the current level, you receive a bonus equal to 1,000 points times the level number.

Directly underneath the radar screen are two additional indicators that show you how many levels have been completed, and how many hits your shields have sustained.

Press the cursor keys to move the aiming crosshair left, right, up, or down. To fire a missile, press the space bar. You can quit the game at any time by pressing Q. In levels 1, 2, 4, 6, and 8 you can view the UFOs through the observatory window. In levels 3, 5, and 7 the sky is obscured by a thick cloud cover, forcing you to guide the missiles by radar alone. The radar screen shows the position of the UFO in relation to your aiming crosshair. Aim with the cursor keys until the red dot is in the center of the radar panel, then fire.

Amiga Version

Converting the original PC/PCjr game to Amiga BASIC was a very easy project. First, we used modems to transfer the PC program text to the Amiga over the phone line. Then we changed a few lines that were obviously unusable in Amiga BASIC (those with KEY and PLAY statements). In less than an hour, after changing about a dozen program lines, we had the PC game running on the Amiga—a testament to the close similarity between the BASICs on both machines.

Though the two programs look very different on the surface, the differences are largely cosmetic. To improve the Amiga program's readability and make it easier to type, we stripped off the line numbers, substituted meaningful labels where needed, and chopped most multi-statement lines into single-statement lines. Line numbers are unnecessary in Amiga BASIC; statements like *GOSUB PrintMessage* and *GOTO MainLoop* are much easier to understand than number-oriented statements like *GOSUB 890*. And in most cases there's little to gain by "crunching" multiple statements onto one line. We made no efforts to speed up the Amiga version, yet because of the Amiga's speedier processor, this program runs much faster than the PC/PCjr game.

If you compare the two programs statement by statement, you'll see that they're still nearly identical. Of course, the Amiga needs SCREEN and WINDOW statements to create a graphics screen equivalent to the original PC/PCjr screen. Since ON KEY and PLAY don't exist in Amiga BASIC, substitutes had to be found there as well (we used INKEY$ to read the keyboard and SOUND for sound effects). But the meat of the program—high-resolution drawing with LINE statements and animation with GET and PUT—is exactly the same.

If you're a PC owner who just bought an Amiga, or an Amiga owner looking for more type-in programs, this project shows how simple it can be to convert programs from IBM BASIC to Amiga BASIC. (Another language which is even more similar to Amiga BASIC is Microsoft BASIC for the Macintosh.) As a general rule, any game that relies chiefly on LINE, GET, and PUT should transfer from the

PC to the Amiga quite easily. Just be sure to set up the right sort of screen at the beginning of the program.

To highlight the similarity between the two versions of BASIC, we did not add many machine-specific features to the Amiga version. However, you may find it interesting to add some extra features of your own. For instance, why nót add voice synthesis to the messages that scroll across the screen? If you have a stereo hook-up, you might want to modify the sound routines to take advantage of the Amiga's stereo sound capabilities. The Amiga version of "Switchbox" (COMPUTE!, March 1986) contains examples of how to do both, as well as other tips on writing games in Amiga BASIC. On a larger scale, you might want to try enlarging the playfield. In the original PC/PCjr version, the game screen is kept quite small to make the game run faster. But Amiga BASIC is fast enough to permit convincing animation within a much larger area.

Program 1: UFO Invasion For IBM PC/PCjr

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
LA 10 REM ---------- INITIALIZE
      VARIABLES ----------
PN 20 SCREEN 1:COLOR 0,0,0:CLS:K
      EY OFF:RANDOMIZE TIMER:PLA
      Y "mb"
BF 30 DIM SH(2000),UFO(200),GD(2
      00),X(30),Y(30),RAOAR(50)
IG 40 L=1:TL=B:LIVES=0:SCORE=0:R
      X=100/15:RY=50/12
EL 50 REM ---------- SET UP SCRE
      EN ----------
IJ 60 LINE (111,51)-(211,159),3,
      B:LINE (111,101)-(211,101)
      ,3
NC 70 LINE (121,101)-(121,159),3
      :LINE (131,101)-(131,159),
      3
AF 80 LINE (191,101)-(191,159),3
      :LINE (201,101)-(201,159),
      3
JP 90 FOR Y=157 TO 103 STEP -2
FF 100 LINE (111,Y)-(131,Y),3:LI
      NE (191,Y)-(211,Y),3
HJ 110 NEXT Y
HL 120 CIRCLE (161,120),16,1:CIR
      CLE (161,120),10,1:CIRCLE
       (161,120),4,1
NO 130 LINE (146,120)-(176,120),
      1:LINE (161,108)-(161,132
      ),1
IB 140 FOR X=145 TO 173 STEP 4:L
      INE (X,135)-(X+4,140),3,B
      :NEXT X
OA 150 FOR X=154 TO 162 STEP 4:L
      INE (X,147)-(X+4,151),3,B
      :NEXT X
EJ 160 LINE (1,0)-(6,0),1:LINE (
      0,1)-(7,1),1:LINE (1,2)-(
      7,2),3
IJ 170 LINE (0,3)-(7,3),1:LINE (
      1,4)-(6,4),1:GET (0,0)-(7
      ,6),UFO
JN 180 LINE (0,0)-(7,7),0,BF
LE 190 LINE (0,3)-(4,3),2:LINE (
      2,1)-(2,5),2:GET (0,0)-(5
      ,5),GD
IN 200 LINE (0,0)-(7,7),0,BF
EM 210 GOSUB 270:B$=" ENEMY ALER
      T":GOSUB 250
BE 220 B$=" UFO INVASION":GOSUB
      250
LA 230 LINE (0,0)-(1,1),2,BF:GET
       (0,0)-(1,1),RADAR
```

The IBM PC/PCjr version of "UFO Invasion" pits you, the lonely defender in an Arctic wasteland, against waves of oncoming robot craft.

```
DP 240 GOTO 320
PA 250 FOR I=1 TO 39:LOCATE 1,I:
      FOR SA=1 TO 20:NEXT SA:PR
      INT LEFT$(B$,40-I):NEXT I
NI 260 RETURN
HK 270 FOR I=1 TO 10
KC 280 FOR P=1000 TO 1900 STEP 2
      5:SOUNO P,.2:NEXT P
OK 290 NEXT I
NN 300 RETURN
JA 310 REM ---------- SET UP KEY
      BOARO ----------
KO 320 DEF SEG=0:POKE 1047,PEEK(
      1047) OR 64
OC 330 ON KEY (11) GOSUB 630:KEY
       (11) ON
IL 340 ON KEY (12) GOSUB 640:KEY
       (12) ON
NE 350 ON KEY (13) GOSUB 650:KEY
       (13) ON
BN 360 ON KEY (14) GOSUB 660:KEY
       (14) ON
JN 370 KEY 15,CHR$(&H40)+CHR$(&H
      39):ON KEY (15) GOSUB 670
      :KEY (15) ON
OE 380 KEY 16,CHR$(&H40)+CHR$(&H
      10):ON KEY (16) GOSUB 680
      :KEY (16) ON
CB 390 REM ---------- START A NE
      W LEVEL ----------
CC 400 B=160:PTS=0:T=0
IG 410 FOR S=15B TO 102 STEP -2:
      LINE (202,S)-(210,S),2:NE
      XT S
IC 420 IF L=3 OR L=5 OR L=7 THEN
       CLR=0:GOTO 440 ELSE CLR=
      1
NA 430 FOR S=1 TO 60:PSET(112+RN
      D*9B,52+RND*4B),INT(RNO*4
      ):NEXT S
EA 440 S=100
FF 450 XU=112+RNO*90:YU=52+RNO*4
      0:IF CLR THEN PUT(XU,YU),
      UFO,XOR
FF 460 XG=160:YG=75:PUT(XB,YG),G
      D,XOR
OJ 470 LINE(142+L*4,136)-(144+L*
      4,139),1,BF
OC 4B0 REM ---------- PERFORM MA
      IN LOOP ----------
CO 490 GOSUB 700 ' MOVE CROSS HA
      IRS
LO 500 GOSUB 550 ' MOVE UFO
BD 510 IF FIRED THEN GOSUB 770:I
      F S=15B THEN 1000
ON 520 T=T+1:IF T>TL THEN B=B-2:
      T=0:LINE(192,B)-(200,B),2
      :IF B=102 THEN GOSUB 1190
       ' CHECK TIME
IH 530 GOTO 490
KK 540 REM ---------- MOVE UFO &
       RADAR ----------
KF 550 IF RND<.1 THEN CXU=RND*10
      -5:CYU=RND*6-3
PO 560 IF CLR THEN PUT (XU,YU),U
      FO,XOR
LM 570 XU=XU+CXU:IF XU>200 THEN
      XU=200 ELSE IF XU<112 THE
      N XU=112
GN 5B0 YU=YU+CYU:IF YU>90 THEN Y
      U=90 ELSE IF YU<52 THEN Y
      U=52
OJ 590 IF CLR THEN PUT (XU,YU),U
      FO,XOR
LF 600 PUT(XR,YR),RADAR,XOR:XR=1
      61+(XU-XG)/RX:YR=120+(YU-
      YG)/RY:PUT (XR,YR),RAOAR,
      XOR
NC 610 RETURN
FE 620 REM ---------- RESPOND TO
       KEY PRESSES ----------
KP 630 CYG=CYG-5:RETURN
JB 640 CXG=CXG-5:RETURN
IL 650 CXG=CXG+5:RETURN
JN 660 CYG=CYG+5:RETURN
OB 670 FIREO=1:RETURN
LO 680 B$=" GAME STOPPED":SCORE=
      SCORE+PTS*100*L:RETURN 13
      90
KM 690 REM ---------- MOVE CROSS
       HAIRS ----------
OO 700 PUT (XG,YG),GD,XOR
LO 710 XG=XG+CXG:IF XG>200 THEN
      XG=200 ELSE IF XG<112 THE
      N XG=112
LO 720 YG=YG+CYG:IF YG>90 THEN Y
      G=90 ELSE IF YG<52 THEN Y
      G=52
KN 730 CXG=0:CYG=0
OG 740 PUT (XG,YG),GD,XOR
NL 750 RETURN
LA 760 REM ---------- FIRE -----
      ------
NP 770 PLAY "L64 T255 BAGFEOC <B
      AGFEDC>"
OJ 7B0 IF CLR THEN PUT (XU,YU),U
      FO,XOR
NG 790 LINE (160,100)-(XG+3,YG+6
      ),2
GG 800 FIREO=0 : LINE (160,100)-
      (XG+3,YG+6),0
IC 810 IF XG+3>XU AND XG+3<XU+9
      AND YG+4>YU AND YG+3<YU+6
       THEN 850
PO 820 IF CLR THEN PUT (XU,YU),U
      FO,XOR
NI 830 RETURN
JE 840 REM ---------- UFO IS HIT
       ----------
EJ B50 PUT (XG,YG),GD,XOR
CP B60 FOR E=1 TO 30:X(E)=XU+RND
      *6+1:Y(E)=YU+RND*6+1:PSET
       (X(E),Y(E)),2:SOUNO 80,.
      1:NEXT E
NG 870 FOR E=1 TO 30:PRESET (X(E
      ),Y(E)):NEXT E
EL BB0 IF CLR THEN PSET (XU+RND*
      6+1,YU+RND*6+1),INT(RND*4
      )
GG B90 XU=112+RNO*90:YU=52+RNO*4
      0:IF CLR THEN PUT(XU,YU),
```

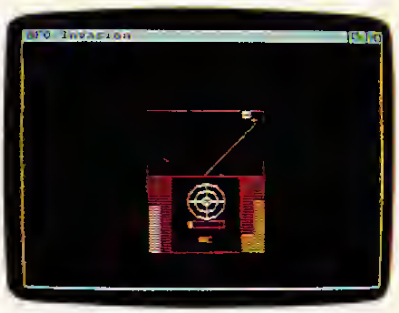
```
      UFO,XOR
KI 900 PUT(XG,YG),GO,XOR
FJ 910 REM ---------- ADD SCORE
       ----------
DG 920 S=S+2:LINE(202,S)-(210,S)
       ,0
HN 930 PTS=PTS+1:GOSUB 970
DF 940 IF 8>130 THEN PTS=PTS+1:G
       OSUB 970
EJ 950 FOR X=8 TO 160 STEP 2:LIN
       E(192,X)-(200,X),0:NEXT X
       :8=160
MP 960 RETURN
BE 970 HX=112:HY=160-PTS*2:IF PT
       S>29 THEN HX=122:HY=160-(
       PTS-29)*2
GL 980 LINE (HX,HY)-(HX+8,HY),1
MF 990 RETURN
BH 1000 REM ---------- ALL UFOS
       DESTROYED - LEVEL COMPLE
       TED ----------
IP 1010 L=L+1:TL=TL-1
BO 1020 FOR P=102 TO 158 STEP 2:
       LINE(112,P)-(120,P),0:LI
       NE(122,P)-(130,P),0:NEXT
        P
BI 1030 LINE(112,52)-(209,99),0,
       BF
LI 1040 SCORE=SCORE+PTS*(L-1)*10
       0
ON 1050 IF PTS<58 THEN 1140
DD 1060 PLAY "O3T120 L8GGL16EEL8
       EL16D+EL8EP8 L16EEL8EL16
       EEL8GL16EGL4FL8DP8"
PO 1070 PLAY "DDL16C+DL8DL16C+DL
       8FP4 L16ED+EL8GP16G16A8A
       8O3P4
KI 1080 B$=" YOU PASSED LEVEL "+
       STR$(L-1):GOSUB 250
EM 1090 PLAY "O3T120 L8GGL16EEL8
       EL16D+EL8EP8 L16EEL8EL16
       EEL8GL16EGL4FL8DP8"
NA 1100 PLAY "ODL16C+DL8DL16C+DL
       4FP8L16CDEGP8L16CDEGP8 L
       16CDE04C4"
JC 1110 B$=STR$(1000*(L-1))+" PO
       INTS BONUS":GOSUB 250
KF 1120 FOR I=1 TO 1000:NEXT I
FH 1130 SCORE=SCORE+1000*(L-1)
DF 1140 IF L>8 THEN 1370
IE 1150 B$=" LEVEL "+STR$(L):GOS
       UB 250
FD 1160 FOR I=1 TO 500:NEXT I
GO 1170 GOTO 400
JD 1180 REM ---------- TIME'S UP
        - TAKE A HIT ----------
MA 1190 X8=XU+4:Y8=YU+6:IF CLR T
       HEN PUT (XU,YU),UFO,XOR
CJ 1195 LINE(X8,Y8)-(112,100),2:
       LINE (X8,Y8)-(210,100),2
KJ 1200 LINE (X8,YU)-(112,52),2:
       LINE (X8,YU)-(210,52),2
FN 1210 LINE(X8,Y8)-(112,100),0:
       LINE (X8,Y8)-(210,100),0
GP 1220 LINE (X8,YU)-(112,52),0:
       LINE (X8,YU)-(210,52),0
BH 1225 IF CLR THEN PUT (XU,YU),
       UFO,XOR
GL 1230 FOR I=1 TO 20
DB 1240 COLOR 0,1
IA 1250 PLAY "L64 T255 8AGFEDC <
       8AGFEDC>"
DJ 1260 COLOR 0,0
HI 1270 NEXT I
PK 1280 PTS=PTS-2:IF PTS<0 THEN
       PTS=0
BE 1290 IF LIVES=3 THEN 1340
GD 1300 FOR X=102 TO 158 STEP 2:
       LINE(192,X)-(200,X),0:NE
       XT X:8=160
BB 1310 LIVES=LIVES+1:LINE (151+
       4*LIVES,148)-(153+4*LIVE
       S,150),2,BF
IH 1320 RETURN
EI 1330 REM ---------- UFOS WIN
       ----------
NN 1340 SOUND 130,0:COLOR 0,1
DO 1350 B$=" GAME OVER":GOTO 139
       0
MJ 1360 REM ---------- SAM WINS
       ----------
KB 1370 B$=" YOU WIN !!!":GOSUB
       1460
LP 1380 GOSUB 1460
KH 1390 GOSUB 250
BE 1400 C$=" SCORE"+STR$(SCORE)
DC 1410 FOR I=1 TO 15:LOCATE 1,I
       :PRINT LEFT$(C$,40-I):NE
       XT I
CH 1420 IF INKEY$<>"" THEN 1420
IN 1430 LOCATE 23,7:INPUT "ENTER
        Y TO PLAY AGAIN:  ";R$
DN 1440 IF R$="Y" THEN RUN
II 1450 END
BB 1460 PLAY "T80O3ML C4G4F16E16
       D16O4C4 O3G4F16E16O1604C
       4"
OP 1470 PLAY "O3G4F16E16F16O4MSC
       4"
KL 1480 RETURN
```

Program 2: UFO Invasion For Amiga

Version by Philip I. Nelson, Assistant Editor

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
<
DEFINT A-Z<
RANDOMIZE TIMER<
SCREEN 1, 320, 200, 2, 1<
WINDOW 1,"UFO Invasion",(0,0)-(3
11,185),20,1<
PALETTE 0,0,0,0<
PALETTE 1,1,1,1<
PALETTE 2,0,1,0<
PALETTE 3,1,0,0<
CLS<
DIM SH(2000),UFO(200),GD(200)<
DIM X(30),Y(30),Radar(50)<
L=1: TL=8: Lives=0: Score=0<
RX=100/15: RY=50/12<
LINE (111,51)-(211,159),3,B<
LINE (111,101)-(211,101),3<
LINE (121,101)-(121,159),3<
LINE (131,101)-(131,159),3<
LINE (191,101)-(191,159),3<
LINE (201,101)-(201,159),3<
FOR Y=157 TO 103 STEP -2<
LINE (111,Y)-(131,Y),3<
LINE (191,Y)-(211,Y),3<
NEXT<
CIRCLE (161,120),16,1<
CIRCLE (161,120),10,1<
CIRCLE (161,120),4,1<
LINE (146,120)-(176,120),1<
LINE (161,108)-(161,132),1<
FOR X=145 TO 173 STEP 4<
LINE (X,135)-(X+4,140),3,B<
NEXT<
FOR X=154 TO 162 STEP 4<
LINE (X,147)-(X+4,151),3,B<
NEXT<
LINE (1,0)-(6,0),1<
LINE (0,1)-(7,1),1<
LINE (1,2)-(7,2),3<
LINE (0,3)-(7,3),1<
LINE (1,4)-(6,4),1<
GET (0,0)-(7,6),UFO<
LINE (0,0)-(7,7),0,bf<
LINE (0,3)-(4,3),2<
LINE (2,1)-(2,5),2<
GET (0,0)-(5,5),GD<
LINE (0,0)-(7,7),0,bf<
LINE (0,0)-(1,1),2,bf<
GET (0,0)-(1,1),Radar<
PUT (0,0),Radar<
GOSUB Siren<
B$=" Enemy Alert "<
GOSUB PrintMessage<
GOSUB Siren<
B$=" UFO Invasion "<
GOSUB PrintMessage<
GOTO NewLevel<
<
<
PrintMessage:<
FOR J=1 TO 39<
LOCATE 1,J<
SOUND 400+(J*10),.1<
PRINT LEFT$(B$,40-J)<
NEXT<
RETURN<
<
<
Siren:<
FOR J=1 TO 10<
FOR P=1000 TO 1900 STEP 55<
SOUND P,.2<
NEXT<
NEXT<
RETURN<
```



This photo illustrates how similar the Amiga version of "UFO Invasion" is to the PC/PCjr game. About 90 percent of the code is identical to the original program.

```
<
<
NewLevel:<
B=160: Pts=0: T=0<
FOR S=158 TO 102 STEP -2<
LINE (202,S)-(210,S),2<
NEXT<
IF L=3 OR L=5 OR L=7 THEN clr=0:
GOTO Mip ELSE clr=1<
FOR S=1 TO 60<
PSET(112+RND*98,52+RND*48),INT(R
ND*4)<
SOUND 1500+INT(RND(1)*1000),.1<
NEXT<
Mip:<
S=100: Xu=112+RND*90<
Yu=52+RND*40<
IF clr THEN PUT(Xu,Yu),UFO<
xg=160: yg=75<
PUT(xg,yg),GD<
LINE(142+L*4,136)-(144+L*4,139),
1,bf<
<
<
MainLoop:<
X$=INKEY$: IF UCASE$(X$)="Q" THE
N Quit<
IF X$="" OR X$<CHR$(28) OR X$>CH
R$(32) THEN Skip<
ON ASC(X$)-27 GOTO Up, Down, Rig
ht, Left, Hit<
GOTO Skip<
Up:<
Cyg=Cyg-5:GOTO Skip<
Down:<
Cyg=Cyg+5:GOTO Skip<
Right:<
Cxg=Cxg+5:GOTO Skip<
Left:<
Cxg=Cxg-5:GOTO Skip<
Hit:<
```

```
Fired=1
<
<
Skip:
PUT (xg,yg),GD
xg=xg+Cxg
IF xg>200 THEN xg=200 ELSE IF xg
<112 THEN xg=112
yg=yg+Cyg
IF yg>90 THEN yg=90 ELSE IP yg<5
2 THEN yg=52
Cxg=0:Cyg=0
PUT (xg,yg),GD
IF RND<.1 THEN Cxu=RND*10-5:Cyu=
RND*6-3
IF clr THEN PUT (Xu,Yu),UFO
Xu=Xu+Cxu
lF Xu>200 THEN Xu=200 ELSE IF Xu
<112 THEN Xu=112
Yu=Yu+Cyu
IF Yu>90 THEN Yu=90 ELSE lF Yu<5
2 THEN Yu=52
IF clr THEN PUT (Xu,Yu),UFO
IP NotPirst THEN PUT(Xr,Yr),Rada
r
NotFirst=1
Xr=161+(Xu-xg)/RX
Yr=120+(Yu-yg)/RY
PUT (Xr,Yr),Radar
IP Fired THEN GOSUB Shoot:lF S=1
58 THEN AllGone
T=T+1
IP T>TL THEN B=B-2:T=0:LlNE(192,
B)-(200,B),2:IP B=102 THEN GOSUB
TakeShot
GOTO MainLoop
<
<
Quit:
B$=" Game stopped"
Score=Score+Pts*100*L
GOTO GameOver
<
<
Shoot:
IF clr THEN PUT (Xu,Yu),UFO
LINE (160,100)-(xg+3,yg+6),2
FOR J=50 TO 1000 STEP 200
SOUND J,.1
NEXT
Fired=0
LINE (160,100)-(xg+3,yg+6),0
IP xg+3>Xu AND xg+3<Xu+9 AND yg+
4>Yu AND yg+3<Yu+6 THEN HitUFO
IP clr THEN PUT (Xu,Yu),UFO
RETURN
<
<
HitUFO:
PUT (xg,yg),GD
FOR e=1 TO 30
X(e)=Xu+RND*6+1
Y(e)=Yu+RND*6+1
PSET (X(e),Y(e)),2
SOUND 820,.1
NEXT
FOR e=1 TO 30
PRESET (X(e),Y(e))
NEXT
IF clr THEN PSET (Xu+RND*6+1,Yu+
RND*6+1),INT(RND*4)
Xu=112+RND*90
Yu=52+RND*40
IP clr THEN PUT(Xu,Yu),UFO
PUT(xg,yg),GD
AddScore:
S=S+2
LINE(202,S)-(210,S),0
Pts=Pts+1
GOSUB Here
IP B>130 THEN Pts=Pts+1:GOSUB He
re
FOR X=B TO 160 STEP 2
LINE(192,X)-(200,X),0
NEXT
B=160
RETURN
<
<
Here:
Hx=112: Hy=160-Pts*2
lF Pts>29 THEN Hx=122:Hy=160-(Pt
s-29)*2
LINE (Hx,Hy)-(Hx+B,Hy),1
RETURN
<
<
AllGone:
L=L+1: TL=TL-1
FOR P=102 TO 158 STEP 2
LINE(112,P)-(120,P),0
LINE(122,P)-(130,P),0
NEXT
LlNE(112,52)-(209,99),0,bf
Score=Score+Pts*(L-1)*100
IF Pts<58 THEN GOTO There
POR J=500 TO 2500 STEP 500
SOUND J,2
NEXT
B$=" You passed Level "+STR$(L-1
)
GOSUB PrintMessage
B$=STR$(1000*(L-1))+" Points bon
us"
GOSUB PrintMessage
FOR J=1 TO 2000
NEXT
Score=Score+1000*(L-1)
IP L=9 THEN PlayerWins
There:
IP L>8 THEN PlayerWins
B$=" Level "+STR$(L)
GOSUB PrintMessage
FOR J=1 TO 1500
NEXT
GOTO NewLevel
<
<
TakeShot:
PUT (xg,yg),GD
IF clr THEN PUT (Xu,Yu),UFO
Xb=Xu+4: Yb=Yu+6
LlNE(Xb,Yb)-(112,100),2
LINE (Xb,Yb)-(210,100),2
LINE (Xb,Yu)-(112,52),2
LINE (Xb,Yu)-(210,52),2
LlNE(Xb,Yb)-(112,100),0
LINE (Xb,Yb)-(210,100),0
LINE (Xb,Yu)-(112,52),0
LINE (Xb,Yu)-(210,52),0
IP clr THEN PUT (Xu,Yu),UFO
PUT (xg,yg),GD
PALETTE 0,1,0,0
FOR K=400 TO 500
SOUND K,.1
NEXT
PALETTE 0,0,0,0
Pts=Pts-2
IF Pts<0 THEN Pts=0
IF Lives=3 THEN UFOgotcha
FOR X=102 TO 158 STEP 2
LINE(192,X)-(200,X),0
NEXT
B=160
Lives=Lives+1
LINE (151+4*Lives,148)-(153+4*Li
ves,150),2,bf
RETURN
<
<
UFOgotcha:
PALETTE 0,1,0,0
FOR J=400 TO 500 STEP 3
SOUND J,.1
NEXT
FOR J=500 TO 400 STEP -3
SOUND J,.1
NEXT
PALETTE 0,0,0,0
B$=" Game Over"
GOTO GameOver
<
<
PlayerWins:
8$=" You win!"
GOSUB WinSound
<
<
GameOver:
GOSUB PrintMessage
c$=" Score"+STR$(Score)
FOR J=1 TO 15
LOCATE 1,J
PRlNT LEFT$(c$,40-J)
NEXT
CleanBuffer:
IF INKEY$<>"" THEN CleanBuffer
LOCATE 23,9
PRINT "Press Y to play again";
X$=""
WHILE X$=""
X$=INKEY$
WEND
IF UCASE$(X$)="Y" THEN RUN
CLS: END
<
<
WinSound:
FOR J=1 TO 2
RESTORE MusicData
SoundLoop:
READ X
IF X<>65535& THEN
SOUND X,1
GOTO SoundLoop
END IF
NEXT
SOUND 550, 8
RETURN
<
<
MusicData:
DATA 550, 500, 450, 400, 350
DATA 550, 500, 450, 400, 350
DATA 300, 350, 400, 450, 500
DATA 65535
<
```

Skyfox For Commodore And Apple

Richard Mansfield, Senior Editor

Requirements: Commodore 64 with a joystick; Apple II-series computer with at least 64K RAM; Apple Macintosh; Amiga with at least 256K RAM (joystick optional). The Amiga version was reviewed.

Some games are all strategy, some are all action, but many of the best games require both forethought and quick reflexes. *Skyfox* is one of those hybrid games, and it's clearly one of the best available for the Amiga. With its many levels of difficulty and player options, virtually anyone will find it challenging and rewarding.

The elements of strategy in this game recall the venerable computer game *Star Trek*. You're the last hope of the Federation asteroid base, the only pilot available. What's more, you've got to fly this experimental jet without sufficient training, and you can't even recall everything this advanced craft can do.

The asteroid base has been attacked by The Enemy, one or more immense motherships which convulsively disgorge wave after wave of tanks and planes. *Their* mission is to destroy the Federation Homebase which houses the Skyfox computer and the only place where you can refuel and recharge your shields. From time to time, you must check with your computer's grid map of the entire asteroid to see where enemy forces are massing. If they manage to get close to Homebase, you should try to take them out. If Homebase is destroyed you can still prevail, but it will be far more difficult.

The action elements of the game are among the best you'll ever see: realistic, realtime graphics; excellent stereo sound; complex air and ground battle scenes. Heat-seeking missiles, laser cannons, enemy tanks and planes, clouds, cockpit controls, Homebase, guided missiles, trees, shrubs, and sky are all vivid and believably recreated using computer graphics in three dimensions. *Skyfox* is more than a game; it's an effective visual and aural simulation.

Ace Of The Base

The simulation is made more rich by the large number of options you have during your struggle to overcome The Enemy: a tactical map; zoom maps of individual sectors; automatic pilot; an installation status report; fuel, speed, and shield indicators; x and y coordinates; a compass readout; forward and rear radar scanners; techniques to move between sky and ground battle; and an altitude indicator. Make good use of these tools and you'll find yourself capable of moving up in rank and attempting some of the more drastic invasion scenarios.

Before an invasion starts, you select one of five skill levels ranging from Cadet through Ace of the Base. Then you choose one of the 15 scenarios. There are 7 training scenarios during which you can work to improve the accuracy of your control over the inertial motion of Skyfox and steel your nerves against the smoke and flame and relentless attack of enemy tanks and planes. There are no Motherships during training, so there is a finite number of attackers. Also, Homebase cannot be destroyed.

When you feel confident that you're ready for the real thing, select a Small, Full, or Massive Invasion. These differ primarily in the number of Motherships active during the game. If you eventually become truly skilled, there are the five ultimate invasions during which multiple Motherships attack using different formations and varying strategies to take out your Homebase. These scenarios are called Halo, Alamo, Advancing Wall, Chess, and Cornered.

The Amiga, with its speed, sophisticated graphics, and quality stereo sound, is an excellent medium for this challenging, vividly realized game. The designers and programmers have outdone themselves in exploiting the Amiga's powerful features and have, in Amiga *Skyfox*, created a simulation which rivals the best computer games available in any medium.

Skyfox *is an exciting action-strategy game that reveals much of the potential of the Amiga's graphics and sound.*

Skyfox
Electronic Arts
2755 Campus Drive
San Mateo, CA 94403 $32.95 (64 version)
$39.95 (all other versions)

The Battle Of Antietam

James V. Trunzo

Requirements: Apple II-series computer with at least 48K RAM; Commodore 64 or 128; or an Atari 400/800/XL/XE with at least 48K RAM. Disk only.

Less than a year before the battle of Gettysburg, a Civil War conflict erupted that became known as "the bloodiest day in American history." In Sharpsburg, Maryland, the battle of Antietam produced more than 22,000 casualties, and it has since been one of the most debated encounters of the Civil War.

The Union army, under the command of General McClellan, outnumbered Robert E. Lee's Confederate forces by more than two to one. Yet throughout the course of the battle, the cautious and indecisive McClellan failed to commit the bulk of his army. Along with a number of other blunders, this turned the day's battle into a

All The Best

APPLE INFORMATION

Take advantage of the spectacular special features of your Apple II-series and Macintosh computers with these bestsellers from COMPUTE! Books.

Advanced Macintosh BASIC Programming

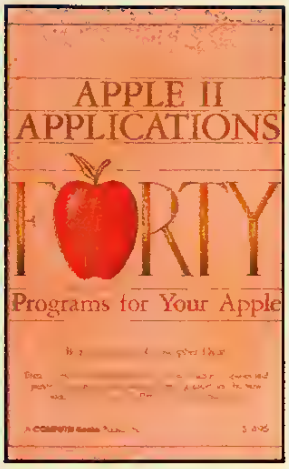
Philip Callippe, 309 pages
A reference guide and tutorial to Microsoft BASIC which shows you how to use the Macintosh's advanced features to create impressive programs. *A disk is also available which includes programs in the book, $15.95 (0300BDSK)*

$16.95 ISBN 0-87455-030-0

Using Your Macintosh: Beginning Microsoft BASIC and Applications

Richard K. Swadley and Joseph Boyle Wikert, 274 pages
Necessary and easy-to-understand information about the revolutionary Macintosh along with clear, easy-to-follow explanations of BASIC. Everything from writing your first statement to creating a finished program. *A disk is also available which includes programs in the book, $15.95 (0211BDSK).*

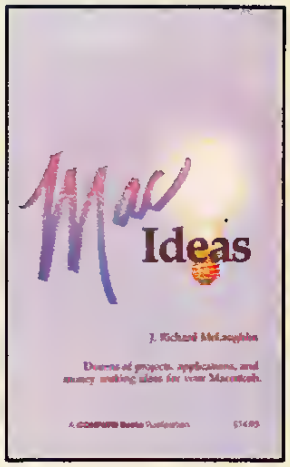
$16.95 ISBN 0-87455-021-1



Apple II Applications: Forty Programs for Your Apple

Brian Flynn and Christopher Flynn, 374 pages
Forty educational and strategy games, business and science applications, and home and personal organizational tools to use on any Apple II-series computer. *A disk package is also available which includes programs in the book. Specify DOS 3.3 or ProDOS, $12.95*

$14.95 ISBN 0-87455-016-5



MacIdeas

J. Richard McLaughlin, 240 pages
More than 100 ways to utilize the Macintosh's powerful graphics capabilities. Beautify everything from personal gifts to correspondence, and learn how to use digitizers to create dazzling graphics.

$14.95 ISBN 0-87455-015-7

Mail this coupon with your payment to COMPUTE! Books, P.O. Box 5038, F.D.R. Station, New York, NY 10150.
Or call toll-free 800-346-6767 (in NY 212-887-8525).

Please send me the following books and disks:

____	**Advanced Macintosh BASIC Programming (030-0), $16.95 each**	____
____	**Advanced Macintosh BASIC Programming Disk (0300BDSK), $15.95 each**	____
____	**Apple II Applications (016-5), $14.95 each**	____
____	**Apple II Applications Disk, DOS 3.3 (0165DSK1), $12.95 each**	____
____	**Apple II Applications Disk, ProDOS (0165DSK2), 12.95 each**	____
____	**Apple II Applications Book and DOS 3.3 Disk Combination package (050-5), $29.95 each**	____
____	**MacIdeas (015-7), $14.95 each**	____
____	**Using Your Macintosh (021-1), $16.95 each**	____
____	**Using Your Macintosh Disk (0211BDSK), $15.95 each**	____

All orders must be prepaid in U.S. funds
Payment enclosed (check or money order)
☐ Charge ☐ MasterCard ☐ Visa ☐ American Express

Subtotal	____
NC residents add 4.5% sales tax	____
Shipping and handling (In U.S. $2.00 per disk or book, airmail $5.00 per item)	____
Total enclosed	____

Account No. ____ Exp. Date ____ (required)
Name ____
Address ____
City ____ State ____ Zip ____
Please allow 4–6 weeks for delivery
Prices subject to change without notice.

36612211

COMPUTE! Publications, Inc. abc
Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies
825 7th Avenue, 6th Floor, New York, NY 10019
Publishers of COMPUTE!, COMPUTE!'s Gazette, COMPUTE!'s Gazette Disk, COMPUTE! Books, and COMPUTE!'s Apple Applications

COMPUTE! books are available in the U.K., Europe, the Middle East, and Africa from Holt Saunders, Ltd., 1 St. Anne's Road, Eastbourne, East Sussex BN21 3UN, England, and in Canada from McGraw-Hill, Ryerson Ltd., 330 Progress Ave., Scarborough, Ontario, Canada M1P 2Z5.

COMPUTE!'s *All New* Apple Applications Special

COMPUTE!'s latest *Apple Applications Special* features in-depth articles and interviews, all the inside news about Apple, clearly written tutorials, software buyer's guides, new product information, and valuable ready-to-type-in programs for all Apple users.

Apple owners find these special Apple issues the most understandable, complete, and valuable resources available today.

PLUS

All the programs in *COMPUTE!'s Apple Applications Special* are also available on a timesaving disk, ready to run on your Apple II, II+, IIe, and IIc. The *Disk* costs only $12.95 (plus $2.00 shipping and handling) and gives you immediate access to all the great programs in this special issue.

Look for the Spring/Summer 1986 issue of *COMPUTE!'s Apple Applications Special* on sale where you buy other COMPUTE! publications, or order directly from COMPUTE!. This special issue goes on sale April 8, 1986.

Send in the attached order card or call toll free 800-346-6767 (in NY call 212-887-8525).

Features

• **Business Applications**

'86 Apple: An Interview with John Scully

A wide-ranging interview with the president of Apple. The company's plans for the coming year, its markets, the new Macintosh, and the viability of the Apple II.

Business Software Buyer's Guide

A buyer's guide to the newest Apple II and Macintosh word processors, databases, spreadsheets, and more.

The Expanding Mac

• **Education**

Apple Rules the Schools

Why does Apple have a lock on educational computing? Comments from teachers, administrators, and Apple.

Computers and the Humanities

Educational Software Buyer's Guide

• **The Expanding Apple**

It's New II

A multitude of new hardware and software for the Apple II line—from color printers to Mac-like software—is evaluated.

Weirdware: Off the Beaten Software Path

Weirdware—out of the ordinary software—can turn the Apple II or Macintosh into a telescope, astrological fortune-teller, baby evaluator, and much, much more.

MacAdds: More for the Macintosh

Applications

• **Utilities and Tutorials**

Windows

Create Macintosh-style windows on any Apple II-series computer. Set window size, open, close, and retrieve information.

Mouse Cursor

A Macintosh BASIC utility for altering the mouse pointer. Design data can be saved, then used in other BASIC programs.

Your Personal Ledger

A complete personal financial application for tracking expenses, income, and assets. Easy to use, and packed with features from report generation to customized category codes.

Personal Publishing With Your Macintosh

Tutorial and guide to using such software as *MacPaint* and *MacWrite* to customize letterheads, cards, banners, and more.

Keynote

• **Education and Recreation**

Lexitron

Entertaining word game where players try to beat the clock while finding as many hidden words as possible.

Backgammon

Play the computer in this classic game. This version observes all the rules of standard backgammon.

Apple Automatic Proofreader

COMPUTE! Publications, Inc.
Part of ABC Consumer Magazines, Inc
One of the ABC Publishing Companies
825 7th Avenue, 6th Floor, New York, NY 10019
Publishers of COMPUTE! COMPUTE!'s Gazette COMPUTE!'s Gazette Disk COMPUTE! Books and COMPUTE!'s Apple Applications

nightmare encounter and possibly prolonged the Civil War by years. Had McClellan been more aggressive, the Confederacy might have been crushed at Antietam and the course of history changed.

"What might have been" is exactly what makes Strategic Simulation's *The Battle of Antietam* such an excellent game. You can choose to follow the exact order of battle, with troops being committed as they actually were during the real fighting, or you can take total control and have all troops put into action from the start of the battle and attempt to change the outcome of this bloody day in American history.

Like all SSI games, *The Battle of Antietam* has been meticulously researched and is a tactical game on a grand scale, incorporating 17 weapon types plus a wide variety of options. The game can be played on an introductory, intermediate, or advanced level; units may be represented by icons or symbols; units may be hidden or visible; and map details include towns, streams, ridges, and bridges superimposed on a square grid that displays four elevations. There are many other options, as well.

Union Frustration

But it's more than just the accuracy and playability that makes this 11- to 15-hour game so special. Perhaps it's the battle itself.

When using the Activation option, troops are not available to the player until the time at which they historically entered the battle. This creates an extremely realistic simulation. In fact, when I tried commanding the Union forces using this option, I've never experienced such frustration. Turn after turn I watched the valiant blue coats charge the Confederate positions, fighting to gain a bridge or a hill. I watched them dissolve before the Confederate artillery, break ranks, and retreat—while a huge Union force sat dormant within striking range of the enemy. I came away with a much better understanding and appreciation of just what had occurred at Antietam—and this is what a computer simulation is all about.

Beyond these features, *The Battle of Antietam* incorporates such factors as fatigue, chain of command, limbering and unlimbering artillery, mounting and dismounting cavalry, line-of-sight targeting (which requires only a touch of the key to highlight all possible targets), and more tactical control than any other game in its class. The game may be played solitaire—with the computer commanding either force—or two players can compete head-to-head and try to match Lee's genius and avoid McClellan's indecision.

SSI has produced dozens of computer war games, gathering praise from many sources. *The Battle of Antietam*, however, may transcend previous efforts and become a true classic.

The Battle of Antietam
Strategic Simulations, Inc.
883 Stierlin Road
Mountain View, CA 94043-1983
$49.95

OnLine! For Amiga

Philip I. Nelson, Assistant Editor

Requirements: Amiga computer with RS-232C modem.

OnLine! is a full-featured telecommunications program that allows any Amiga to communicate with remote computers, bulletin boards, and commercial information services such as CompuServe. Since *OnLine!* takes full advantage of the Amiga's graphics-oriented operating system, the program is intuitive and convenient to use. In most cases, selecting an option is as simple as moving the mouse pointer to the desired menu item. But don't confuse ease of operation with a lack of features; this program offers a wide range of options, making it suitable for serious applications as well as recreational use.

For most home use (calling an information service, for instance), you'll want to use the default TTY, or dumb terminal configuration. But you can also choose from three popular DEC terminal modes (VT–102, VT–100, and VT–52) or ANSI emulation. The default window—with a status display line at the top, screen borders, and a sizing gadget at the lower-right corner—has room for a 79-column × 22-line text area. Other display options include a borderless 80 × 23 window, which removes the sizing gadget but leaves the status line in place, and a full 80 × 24 window which has neither a status line nor a sizing gadget.

The most novel display feature is the split or *chat* window, which is designed for realtime electronic conferencing (like the CB service on CompuServe). On many terminal programs, realtime conferencing is a very confusing business. Since your own keystrokes are intermixed with incoming characters, it's very difficult to keep track of what you're typing. By echoing only your keystrokes in a separate win-

dow, *OnLine!*'s chat feature eliminates the confusion.

Unlike some early Amiga software that completely takes over the machine, *OnLine!* is clearly designed to exist in a multitasking environment. In all configurations except the 80 × 24 window, you can use sizing and/or depth gadgets to gain access to the Workbench or other windows. This welcome feature makes it possible to perform other tasks while the terminal remains active. For instance, you might want to open a new CLI window to check whether a disk has enough space to hold a file that you've captured.

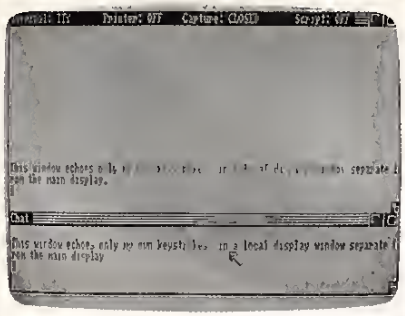
Flexibility

Few things are more frustrating than establishing a communications link only to find that the computer at the other end of the line requires a protocol that your software can't handle. *OnLine!* goes to considerable lengths to provide control over all the parameters you need, without forcing you to specify settings more often than necessary. When you first run the program, it defaults to the configuration used by most commercial information services: 1200 bps (bits per second), 7-bit word length, even parity, and a stop bit of 1. But these parameters (and many more) are easily changed via onscreen menus.

Once you've chosen new settings, you can save them in a *terminal file*, which also includes display choices, phone numbers (for an autodialing modem), and macrokey definitions (see below). Terminal files are a real boon to anyone who calls more than one service regularly. Instead of reconfiguring the program manually each time, you need only set the parameters once for each service and save them in a terminal file. After that, you simply select the desired terminal file from a menu. When *OnLine!* loads the file, it configures the display window, sets all the necessary parameters, and even dials the number for you automatically.

It's easy to see how this sort of automation speeds up and simplifies the process of getting online. Going one step further, you can also customize the way in which the program boots up. Whenever you run *OnLine!*, it looks for a special file named *OnLine!.trm*. If the disk contains a terminal file of that name, the program comes up with the settings specified in the file, and dials the phone number if one is included.

You can also save time by creating a custom *macrokey* definition for one or more of the Amiga's ten function keys. Once a macrokey has been defined, it sends as many as 64 characters to the serial port with only one keypress. In the simplest case, you might program a



OnLine! is a convenient, professional-quality telecommunications program for Amiga computers. This screen shows the chat window feature designed for real-time teleconferencing.

key to transmit a commonly used command such as BYE or GO AMIGAFORUM. By including control codes and linking together more than one macrokey, it's possible to create much more elaborate one-key sequences.

Unlike some terminal programs, *OnLine!* has no separate phone book as such. Instead, two phone numbers (a primary number and one alternate) can be stored as part of each terminal file. If you need more than two numbers for a certain service, you could store additional numbers in macrokey definitions, which also become part of the terminal file. The autodial feature lets you set the number of times to redial the primary and alternate numbers before giving up. The default number of retries is zero, meaning that if the primary number isn't answered within 30 seconds, *OnLine!* dials the alternate number (if one is supplied) or simply hangs up.

If you've ever had to write a program to transfer data files from one computer to another, you know that character translation, while extremely simple in theory, can soak up a lot of programming time in practice. *OnLine!* lets you edit any of its seven 256-byte character-translation tables (which relate to screen, keyboard, printer, and serial input/output) simply by calling the table from a menu and editing the character values onscreen. This makes it easy to do character translations or filter out undesired characters for various purposes. When streaming input to a printer, for instance, you can check for certain characters which might be interpreted as control codes, producing unwanted results.

Automation

Perhaps the most advanced feature of *OnLine!* is its ability to execute *scripts*. A script file is simply a collection of commands stored in a text file on disk (similar to a batch or script file in AmigaDOS). When you load a script file, *OnLine!* automatically performs all the commands found in the file. In other words, the script feature is actually a mini-language interpreter; you can write simple programs, store them in disk files, and execute them whenever you like. This powerful capability makes it possible for the system to carry out an elaborate series of actions without any supervision on your part.

To illustrate what a script can do, say that you want the program to wait until 3 a.m. (when rates are low), dial up a fictional information service called ChompuSerf, log on to the service, enter Data Library 3 in the area called Amigashop, download a file named EXAMPLE.BAS, log off the service, hang up the phone, and save the captured file to disk. Your script file might look something like this:

```
WAIT UNTIL 03:00
REPLY "ATDT 1 919 555 1212"
WAIT DELAY 50
REPLY "^"
WAIT DELAY 5
REPLY "^"
WAIT STRING "Host:"
REPLY "CIS"
WAIT STRING "User ID:"
REPLY "55555,1212"
WAIT STRING "Password:"
REPLY "BUZZWORD"
WAIT STRING "your choice!"
REPLY "go amigashop"
WAIT STRING ":"
REPLY "DL3"
WAIT STRING ":"
REPLY "DOW"
WAIT STRING ":"
CAPTURE OPEN 100
REPLY "EXAMPLE.BAS"
WAIT STRING ":"
CAPTURE CLOSE
REPLY "BYE"
WAIT DELAY 5
OFFLINE
CAPTURE SAVE "EXAMPLE.BAS"
```

The first command in this script causes *OnLine!* to wait until the system clock equals 03:00, or 3 a.m. (of course, it's your responsibility to set the time correctly at the beginning of the session). The next command calls ChompuSerf by sending a Hayes-format autodial command to the modem. The next two REPLY commands simulate the process of pressing RETURN twice. The following WAIT STRING commands cause the program to pause until a particular character string is received. Each REPLY command sends a character string, so by REPLYing to prompts as needed, we move to the Amigashop section of ChompuSerf, enter Data Library 3, and download the file EXAMPLE.BAS. The CAPTURE OPEN command opens the ASCII capture buffer, specifying a buffer length of

100K. When the capture is complete, we log off ChompuSerf (REPLY "BYE"), hang up the phone (OFFLINE), and save the captured file to disk with CAPTURE SAVE.

The example script is actually quite primitive compared to what *OnLine!*'s command set allows. More advanced commands such as IF, WHEN, ASK, JUMP, SKIP, and ABORT permit the script to test for certain conditions, branch to other parts of the script program, and interact with the user to a certain extent. The DO command even lets you load and execute a second script file from within the first.

Writing an automated script like the example shown here requires that you know in advance exactly what the remote system will send in the way of prompts and what you must supply as responses. The simplest way to glean such information is to note each prompt/reply sequence on paper as you go through a typical session. Once that's done, you can write the script file using the ED system editor or a word processor.

But that process takes time and multiplies the chance for errors. *OnLine!*'s *learn mode* automates the process of creating script files by letting you capture the relevant information on the fly. In learn mode, the program automatically records the most recent prompt as well as your last reply, giving you a chance to edit each string on the spot and insert additional commands before adding it to the script file. At the end of a session, you should have a script that requires little or no extra editing.

Transfer Options

OnLine! offers several options for capturing or sending data files, including ASCII capture, standard XMODEM protocol, XMODEM with CRC (*cyclic redundancy checksum*) error-checking, and HVP (Hayes verification protocol). The timing requirements for standard XMODEM are relaxed somewhat to facilitate communications via packet-switching networks. Though it's not supported by every information service, CRC error-checking improves the reliability of XMODEM transfers.

One headache that confronts Amiga owners concerns XMODEM transfers of executable machine language files. Since the XMODEM protocol always sends a file in even 128-byte chunks, any file that doesn't divide evenly by 128 is padded with extra characters when you download it with XMODEM. If you try to load and run a padded file, AmigaDOS notices the padding, concludes that the file is not executable, and refuses to run it. Chopping off the padding is a simple matter from BASIC, but the file is useless until that's done. So this problem adds just one more layer of aggravation and delay to the process of getting someone else's program to work on your computer.

It's worth noting that the padding problem applies only to XMODEM transfers—more specifically, to XMODEM transfers of executable machine language files or other binary files for which exact file length is critical. It shouldn't affect text that you save from the capture buffer, or ASCII text files (including BASIC programs in ASCII form) downloaded with XMODEM. Of course, the padding problem isn't unique to *OnLine!* or any other terminal program. It's a consequence of the way that XMODEM and AmigaDOS treat certain files, and occurs with any Amiga terminal program that supports XMODEM.

OnLine! does not contain any feature to help you chop executable files downloaded with XMODEM. However, it does support HVP protocol (compatible with *Smartcom*) which can transfer executable files without padding. The only problem with HVP, or any protocol other than standard XMODEM, is that not everyone uses it. (Perhaps the best solution is for programmers to pad their executable files before uploading them to public bulletin boards.)

Confusing Manual

While the *OnLine!* instruction manual is fairly complete, it is disorganized. All the information is there—someplace—but it's not always easy to find. Despite the manual's length of 100 pages, there is no index. Fortunately, documentation is less important for a menu-oriented program of this type, which displays nearly every option onscreen. Many people will be able to use *OnLine!* without glancing at the manual. But some important program features—learn mode, for instance—don't appear in the menus at all.

On the whole, however, *OnLine!* is a very impressive package with the look and feel of a finished, professional product. It's convenient, reliable, and well-integrated with the Amiga's personality. Another plus is the quality of customer service. The authors (Micro-Systems Software, Inc.) offer technical support in two different forms: on voice lines during regular business hours, and on their own 24-hour, 7-day BBS. I found that questions to the customer BBS were answered very promptly.

OnLine!
Micro-Systems Software, Inc.
4301-18 Oak Circle
Boca Raton, FL 33431
$69.95

Hippo Computer Almanac For Atari ST

George Miller
Assistant Technical Editor

Requirements: Atari ST computer with at least one disk drive. Printer optional.

Do you know how many ounces are in a liter? Quick, what time is it in Moscow? What's the zip code for Denver? Who won the Super Bowl in 1974?

No, we're not playing another version of Trivial Pursuit. These are questions you can answer in seconds with the *Hippo Computer Almanac* for the Atari ST, a valuable information resource that points the way toward a new generation of intelligent software.

The higher processing speeds and greater disk capacities available with the new generation of personal computers are making possible more powerful and sophisticated programs. For example, much larger databases are becoming available for use in the home. Although the *Hippo Computer Almanac* is not yet in the class of an encyclopedia on a CD-ROM, it is loaded with information. Over 35,000 pieces of information, in fact, according to Hippo.

It Understands English

Like any good almanac, this electronic repository contains information on such general topics as history, geography, sports, languages, science, awards, and units of measure.

Perhaps the best feature of all is that you communicate with the program by typing plain English sentences. A parser routine swiftly evaluates your query, and the program usually retrieves the information in less than ten seconds. If the almanac doesn't know the answer to a question, there's no cryptic comment or error message. The screen simply displays, "I don't know."

Of course, even with a first-rate parser, there are always going to be occasions when the program won't follow your questions. However, the *Almanac* does have the ability to find the closest match to any request, and it tries to satisfy any query.

If, after several attempts, you still can't make the program understand your question, just type HELP. Online help is always available in all categories. The help screens are easy to understand and even offer sample questions illustrating the format for communicating with the program. As your familiarity with the *Almanac* increases, you'll

learn how to communicate in the least number of words. For instance, "Time London" yields the time of day in London, England, eliminating the need to type "What time is it in London, England?"

A Personalized Almanac

You can also customize your version of the *Almanac*. For instance, it's easy to set up the database so the program knows where you are geographically. This makes it possible for the *Almanac* to calculate time zone differences and mileages between your home town and distant lands. You can also use the "remember" command to store important personal information in the *Almanac*, such as birthdays, anniversaries, and phone numbers.

The *Almanac* is easy to use without extensive instructions. In fact, a single information sheet is provided instead of a manual. There is also an easy-to-use print option that lets you make hard-copies of anything you call up.

Browsing through the *Hippo Computer Almanac* is fun. It's an engaging program that entertains at the same time that it offers a useful database of information.

Hippo Computer Almanac
Hippopotamus Software, Inc.
985 University Avenue, Suite 12
Los Gatos, CA 95030
$34.95

Zoomracks For Atari ST

Arthur Leyenberger

Requirements: Atari ST computer with disk drive. Printer optional.

Zoomracks by Quickview Systems is a powerful, easy-to-use database manager that lets you keep track of lists, names, addresses, notes, schedules—almost anything you can think of—in a unique and interesting way. What's unique about the program is the concept of the "rack."

Consider a familiar timecard rack—the vertical holder that sits next to the time clock and holds employee timecards. The first line of each card is always visible. You can remove any card to examine its contents. Cards can also be inserted or moved into other slots in the same or adjoining racks. Cards in the racks are typically in the same form (timecards), but contain different information, such as names and hours worked. They may be arranged

in some order, such as by name or employee number.

This describes the visual metaphor upon which *Zoomracks* is based—the card rack. It is a familiar concept and translates well to the computer. When you choose a card from the rack in order to see its contents, you notice that it has several fields, each of which shows the top line of information just as all the cards appear. Each field can be pulled up to expose as much as three pages of information.

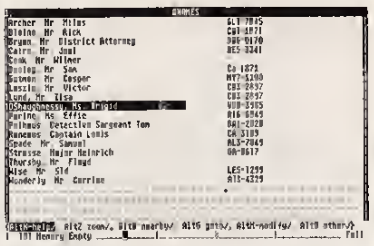
Stretching The Rack

The way your information is organized is always visually obvious because the screen shows as many as ten racks at once. The number of cards in each rack is limited only by the amount of computer memory, and the racks grow or shrink as required. If your rack is too large to fit on the screen, it can be scrolled. Or you can search for the card or field you're looking for. The cards in one rack can be sorted by any field, and each card can have up to 29 fields.

Zoomracks offers three different field types: short fields, text fields, and columns. Short fields are similar to those found in traditional database programs. One field is displayed at a time on each line. Text fields are used for multiple notes. The document (your notes) is displayed across the entire width of the screen on consecutive lines. Finally, the column field is used for spreadsheet-type information—for example, sales orders.

Rack formats can be inserted and deleted by moving fields; cards and fields can be copied between racks; you can do simple word processing, since any field can be up to 250 lines long; and the cards and racks can be printed in many different formats.

One interesting feature of *Zoomracks* is its macro capability. Macros are any series of *Zoomracks* commands that are strung together and issued at once by a single keypress. You can have one rack with up to 26 macros. There are several sample macros provided, and



Zoomracks uses a unique visual metaphor to let you organize and retrieve information.

one serves as a tutorial for the program.

Mail-Merge Feature

The program disk contains several sample racks as well. One sample which is useful for more than just learning about *Zoomracks* is a mail-merge template. One card within a rack serves as a form letter, and the card-merge macro can be used to print out a rack of cards consisting of names, addresses, and salutations. With a little imagination, you can develop all kinds of applications by using macros.

All in all, *Zoomracks* is a practical and even a fun way to keep track of various types of information. With its visual interface and zooming feature, you can always visualize your data as you want—from a broad overview of the whole database to a specific detail of a single field. Help is available at any time and the menus are straightforward. As you get more experienced, you can use the commands instead of the menus. In either case, there are few rules to follow and few limitations.

Zoomracks is a well-done program and a unique concept. If you need a database manager and want to get up to speed as quickly as possible, *Zoomracks* is an excellent choice.

Zoomracks
Quickview Systems
146 Main Street
Los Altos, CA 94022
$80

Stickybear Learning Games For Apple And Commodore

Karen G. McCullough

Requirements: Apple II-series computer with at least 48K RAM and a disk drive. Joystick optional. Commodore 64 version scheduled for release by this summer.

With their Stickybear series, Optimum Resources and Weekly Reader Software have developed a reputation for producing software that is reliable, educational, and entertaining. They maintain those high standards with three new releases: *Stickybear Typing, Stickybear Town Builder,* and *Stickybear Spellgrabber.*

Typing is an application ideally suited to computerized instruction—it's an area where the computer can do a better job than traditional methods of teaching. A good typing tutor program provides immediate feedback—both aural and visual—for incorrect keypresses, and allows a student to progress automatically through levels as each is mastered, rather than dictating progress with a schedule or lesson plan.

Stickybear Typing does all this and more. Each of the program's 30 levels introduces the student to the keys covered in the lesson, then offers practice using them. The lower half of the screen displays the keyboard; as keys are highlighted one at a time, the student must press the corresponding key on the computer's keyboard. A correct keypress prints the letter at the top of the screen. Incorrect keypresses make a low "bloop" sound, and the letter doesn't appear. At the end of each two screens of typing practice, the student gets a progress report which shows the starting level, current level, number of words typed per minute, number of errors, and corrected words per minute.

A Typing Game

Another section of the program—Stickybear Thump—allows typing practice in the form of a game. Stickybear and a robot throw things at each other while the player copies lines of letters displayed on the screen. The robot throws boxes at preset intervals; each time a line is completed, Stickybear throws a ball at the robot. The faster you type, the more balls Stickybear throws, the more points you get, and so on.

A third section of *Stickybear Typing*, the Stickybear Stories Module, provides typing practice of a more practical sort—copying amusing stories, paragraphs, and jokes.

Stickybear Typing has a number of nice features. Up to 25 names can be stored on the disk with current level information for each person. The sound can be toggled on and off, as can a hands display which illustrates proper finger placement on the keyboard. In two sections of the program, you can choose either typewriter mode (you must press RETURN at the end of each line, and you can't backspace to that line) or word processing mode (free-style typing).

Although *Stickybear Typing* is intended primarily for children, it can be used by adults just as effectively. We found only one problem with the program: A decent typist can outrun it. Particularly in the game sections, frustrating errors can occur as the program drops letters which are typed too quickly. However, most students won't be fast enough to experience that problem, at least at first.

Stickybear Typing *offers several ways for youngsters to sharpen their keyboard skills (Apple version).*

Build A Town

Stickybear Town Builder, for children ages six to nine, lets the youngsters build their own towns on the screen, drive through them with a small keyboard- or joystick-controlled car, hunt for hidden keys, and learn some elementary map-reading skills in the process. Towns can be saved and loaded again later, or you can use one of three towns provided on the disk. The graphics are attractive, and the program is easy enough to be used by children even younger than six. But children at the older end of the suggested age range may not find the program challenging enough to hold their attention for long.

If your child needs work on spelling, *Stickybear Spellgrabber* might be the answer. Three different games help a child learn selected word lists. All three games are fun, challenging, and really can help with spelling drills. A nice feature of the program allows you to enter your own spelling list or use one of the four lists included (keyed to grades 1–4). Stickybear can be controlled with either keyboard or joystick. While the joystick is slightly easier to use, both require practice to master. Unlike *Town Builder,* all three games are difficult enough to be challenging even to nine- or ten-year-olds, as well as educational.

Stickybear Typing
Stickybear Town Builder
Stickybear Spellgrabber
Weekly Reader Family Software
245 Long Hill Road
Middletown, CT 06457
$39.95 each (Apple versions)
$29.95 each (64 versions)

Kennedy Approach For Commodore And Atari

David and Robin Minnick

Requirements: Commodore 64 or 128 (in 64 mode); or an Atari 400/800, XL, or XE with at least 48K RAM. Disk drive and joystick also required. The Commodore version was reviewed.

It's 10:53 a.m.

You're in the midst of your second shift as an air traffic controller. Six flights await your clearance for takeoff. Five more are waiting to land. Compounding your headache are a thunderstorm approaching from the west and the Concorde approaching from the east.

Suddenly you hear, "This is United 101. Emergency! Eight minutes fuel!"

The Concorde moves at eight miles every minute. Within two minutes the planes will be at a point of intersection. Unless United 101 gets on the ground fast, lives will be lost.

Your palms begin to sweat.

"United 101. Turn left, heading 90 degrees. Descend to 3,000 feet. Air France 314. Hold right at VDR at 5000 feet."

Oh no! you think, staring at the screen. *I forgot Delta 626 coming in at the same altitude!*

The conflict buzzer sounds.

Your spouse looks up from the couch. "Could you please turn that thing down?"

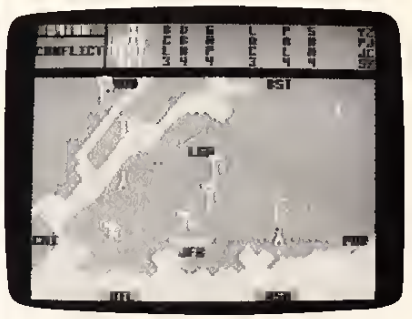
It's Just A Simulation

This is *Kennedy Approach,* an air traffic control simulation from Micro Prose. It puts you in the seat of an air traffic controller in one of five U.S. cities. Each airport presents you with skill levels ranging from 1 (Atlanta—a challenging beginning) to 5 (New York City—no margin for error).

In *Kennedy Approach,* you work a shift of approximately ten minutes real-time, longer at the higher levels. At the end of your shift, your performance is evaluated and you're promoted, given a bonus, or fired. Additional options let you continue your career, see an instant replay, save your shift to resume playing later, or return to the main screen.

It's only a simulation, a game, you tell yourself between shifts—but the sweat on your palms when you play *Kennedy Approach* is quite real.

Keyboard or joystick controls are used to establish contact with a plane. Then the joystick is used to change its heading and/or altitude. A push of the fire button prompts an exchange of dialog between you and the pilot. Probably the most delightful feature of the program is the use of digitized voices for this exchange. This is software-driven speech synthesis from Electronics Speech Systems. The dialogs have the quality of genuine "black box" air traffic recordings.



Keeping the friendly skies friendly is a frenzied job in Kennedy Approach, *an air traffic control simulation (Commodore 64 version).*

The graphics overall are very good, particularly the thunderstorms, but a few effects require getting used to. The one representing a plane's location is somewhat confusing, and it's difficult at first to decipher the display of flight plans. Both these problems are conquered by familiarity.

Some Minor Quirks

There are a few quirks in *Kennedy Approach.* Planes start to wrap around the screen, a sight which can be disconcerting to the newly hired controller. Routing flights into a holding pattern is a lip-biting maneuver, as this requires you to press the fire button at the right moment while commands are sequentially displayed in the command line. This is the most difficult task in the program, and it seems that it could be accomplished more easily.

Another oversight is that *Kennedy Approach* lacks a disk directory function for selecting which shift to retrieve.

The instruction manual is superb in providing information about the air traffic control aspects of the simulation. This technical information allows even the beginner to feel familiar with the new environment. One small flaw, though: At one point the manual directs you to a nonexistent Section VI, leaving you to your ingenuity and experience to discover how to instruct the pilot to climb to the desired altitude at takeoff. (This is corrected in later editions of the manual. Users with early manuals should refer to B-3 instead of Section VI.)

Despite these small problems—they're the only ones we found and are

minor compared to the whole package—*Kennedy Approach* is a fascinating, well-designed simulation for someone who wants to get a taste of what air traffic controllers do all day (and night). More simulation than game, it still elicits game-type responses. If you judge a game by how it affects your psyche, by how excited you get, and by how nervous it makes you, *Kennedy Approach* gets a clammy hands rating of 9 out of a possible 10.

Kennedy Approach
Micro Prose Software
120 Lakefront Drive
Hunt Valley, MD 21030
$34.95

Sideways Text For Atari

Bill Morris

Here's a short machine language routine that converts your lowercase letters (a–z) *to uppercase* sideways *letters. Why? Well, it's so short that it's worth typing in just to see the amusing effect, but it's also useful for labeling charts and designing one-of-a-kind title screens. The program works on any Atari 400/800, XL, or XE.*

Wouldn't it be nice to have sideways letters that could be displayed anywhere on a GRAPHICS 0 screen? Imagine the interesting title displays you could add to programs. Or, from a more practical standpoint, sideways letters could be more than just a show-off effect for charting programs—they could become a necessity.

One way to get sideways letters is to spend a couple of hours with graph paper or a character editor to redefine the lowercase character set. But that would be the hard way. Such a laborious task is best left to a labor-saving device such as your Atari computer.

The program below contains a machine language routine that decides where in memory to place the new character set, relocates the set to that area, changes the character base pointer, erases the lowercase alphabet, and replaces it with uppercase letters that are rotated 90 degrees to the left.

You might notice that the machine language routine doesn't contain any data to define what the sideways letters should look like. Instead, it actually flips each letter mathematically before relocating it in memory. It does all this in about one second and takes up less space in your BASIC program than would the DATA statements alone if you were doing it the hard way.

Sideways Text In Action

If you want to see sideways text on your own computer screen, just type in the program, save a copy on disk or tape, and then run it. What you'll see is the word SIDEWAYS displayed in GRAPHICS 0 actually turned sideways. Next to this you'll see the word TEXT in normal letters.

Everything appears on a light screen with dark characters. On the normal default screen of white letters on a blue background, the sideways text can be hard to read, so dark letters are preferable. Also, for charts, you might want to blank out the screen borders by adding this line:

```
95 POKE 712,PEEK(710)
```

Lines 40–90 POKE the machine language routine into memory page 6, but once the routine is executed, you can reuse this memory for some other purpose without affecting the sideways text. It stays sideways until you press SYSTEM RESET.

Sideways Text For Atari

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
IK 10 ? CHR$(125)
JA 20 PDKE 559,0
NF 30 GDSUB 20000
MJ 40 PDKE 559,34
BE 50 X=USR(1536)
FM 60 PDKE 710,158:PDKE 709,
      0
FN 70 ? " s"
GE 80 ? " y"
EN 90 ? " a"
NA 100 ? " w TEXT"
HK 110 ? " e"
HK 120 ? " d"
IA 130 ? " i"
IL 140 ? " s"
BJ 150 ? "  "
GD 160 END
BL 20000 FOR A=1536 TD 1715:
         READ B:PDKE A,B:NEX
         T A:RETURN
MC 20010 DATA 104,165,89,56,
         233,4
BK 20020 DATA 141,244,2,133,
         205,169
BK 20030 DATA 224,133,207,16
         9,0,133
BB 20040 DATA 204,133,206,16
         2,0,160
BG 20050 DATA 0,177,206,145,
         204,200
IB 20060 DATA 208,249,230,20
         5,230,207
OD 20070 DATA 232,224,4,208,
         238,32
FL 20080 DATA 167,6,160,0,16
         9,0
IC 20090 DATA 145,204,200,19
         2,216,208
JB 20100 DATA 249,32,167,6,1
         69,0
LM 20110 DATA 141,186,6,141,
         182,6
MB 20120 DATA 141,183,6,169,
         8,141
MF 20130 DATA 184,6,141,185,
         6,174
CJ 20140 DATA 182,6,169,128,
         141,181
GB 20150 DATA 6,172,183,6,16
         9,1
LP 20160 DATA 141,180,6,189,
         0,225
ID 20170 DATA 45,180,6,205,1
         80,6
PI 20180 DATA 208,8,177,204,
         24,109
CG 20190 DATA 181,6,145,204,
         173,180
HI 20200 DATA 6,10,141,180,6
         ,200
CC 20210 DATA 204,185,6,208,
         224,173
IN 20220 DATA 181,6,74,141,1
         81,6
CB 20230 DATA 232,236,184,6,
         208,203
MD 20240 DATA 173,184,6,141,
         182,6
IL 20250 DATA 141,183,6,24,1
         05,8
MD 20260 DATA 141,184,6,141,
         185,6
ND 20270 DATA 238,186,6,173,
         186,6
DA 20280 DATA 201,27,208,167
         ,96,173
IF 20290 DATA 244,2,24,105,3
         ,133
OD 20300 DATA 205,169,0,133,
         204,96
```

Loading And Linking Commodore Programs

Part 4: Overlaying

Jim Butterfield, Associate Editor

This installment of Jim Butterfield's series on loading and linking Commodore programs talks about overlays—a technique that allows a program to call in additional subroutines and other data. The principles apply to most Commodore computers, including the 64, 128, VIC-20, PET, Plus/4, and 16.

There are three major ways of connecting Commodore programs together. *Chaining* allows several programs to perform a job, each program continuing the work that a previous program began. *Linking* enables one program to call up another, with the new program starting fresh on a new task. *Overlaying* allows a main program to call in supplementary material such as machine language subroutines, data tables, or display information. This article discusses overlay programming techniques. (Though the example programs are designed for a disk drive, you should be able to change most of them to work with tape by replacing ,8,1 with ,1,1.)

In some situations a computer program may need extra pieces of information to perform its task. The extra material may be one or more programs (often machine language), or it could be pure data. Data can be of several types: information, display screens, character sets, sprite shapes, or whatever. The difference between overlaying and chaining or load linking is that the main program stays in memory at all times, calling up the modules it needs.

Why Overlay?

The classic reason to overlay programs is so that a main program can call up a machine language module to do a specific job. This permits you to keep a library of special programs on disk and call in each program as it is needed. For example, you might bring in one machine language program to scan through a file, searching for information; another might be used to display data neatly on the screen; yet another module could be called to handle printer output, and so on. In the simplest case, only one program module is brought in at a time, and a certain section of the computer's memory is set aside to hold the current program. This lets you run programs which are, in effect, much larger than the amount of memory in your computer.

One obvious use for this technique is to bring in a series of attractive high-resolution graphics screens. Since each hi-res screen requires 8,000 bytes of memory (with more needed for color information), it's not practical to keep more than one or two in memory at a time. But a disk can hold the data for many hi-res screens. By calling in each screen only when it's needed, you can display dozens of hi-res pictures in the course of a program run.

The same factors apply to other sorts of data, too. For instance, a program could use many different sprite shapes as it runs. Sprite-animated figures could change from bicycles to cars, and later to horses, elephants, or boats as a schedule of race events progresses. All that's required is to replace one set of sprite shapes with a new set by means of overlaying.

Alternate character sets also require extensive amounts of data, usually thousands of bytes for each different set. If you want to switch from Roman (the characters you're reading right now) to Greek, Arabic, Hebrew, Russian, or whatever, simply haul in each new character set as you need it.

Breaking The Chain

Before you overlay information, you must set aside space to hold it. This isn't a new requirement: Regardless of where the data comes from, it's always necessary to allocate room for sprite shapes, hi-res

screens, machine language programs, and so on. So we won't repeat the familiar methods of setting aside memory for such purposes.

Let's work through the sequence of events that occur when you bring in an overlay module. Keep in mind that the BASIC program itself is not replaced—the program is still present and running.

The first step is for the BASIC program to load the desired module with a command like LOAD "MODULE",8,1. (The ,1 at the end of the LOAD command is needed on most Commodore computers to specify a *nonrelocating* load—one that loads the file back into the exact part of memory from which it was saved.)

Here comes the tricky part. When the load is complete, the computer thinks that it has performed a *chain*. It concludes (wrongly in this case) that the old BASIC program has been replaced by a new one. None of the program's existing variables are erased or changed, but *the computer reruns the BASIC program from its first line.*

This phenomenon isn't a bug; it's simply what the designers intended to happen whenever you LOAD from within a BASIC program. However, it raises a puzzling problem for beginners. If you write a program that begins with the line 10 LOAD "MODULE",8,1 and run it, here's what happens. The MODULE file is loaded. Then the program reruns, beginning at line 10. So MODULE is loaded again. Then the program reruns again, loading MODULE again, which causes another restart, and so on. Until you press RUN/STOP, the program continues forever.

Fortunately, there's an easy solution. Because LOAD from within a program doesn't destroy existing variables, we can change a variable when the load occurs and use it to branch around the LOAD command when the program restarts. It's like building a bypass around the LOAD after the overlay is complete. Take a look at this program fragment:

```
10 IF A=1 GOTO 40
20 A=1
30 LOAD "MODULE",8,1
40 REM PROGRAM CONTINUES...
```

Let's trace what happens when this program runs. The first time it's run, the variable A is equal to 0 (it hasn't been defined yet). So the IF test in line 10 (which tests for the condition A=1) fails, and we don't branch to line 40. Instead, the program proceeds to the next line. Line 20 then makes A equal to 1. Line 30 loads the MODULE file to wherever it's going in memory. At this point (the end of line 30), the program goes back to the first statement. This time the IF test is true (A *is* equal to 1), so we branch to line 40. The program continues without getting caught in an endless series of loads. You could also condense the whole operation into one program line:

```
10 IF A=0 THEN A=1: LOAD "MODULE",8,1
20 REM PROGRAM CONTINUES
```

This example combines the IF test, the setting of A to 1, and the LOAD command all in one line. Another option is to replace line 10 of the original example with 10 ON A GOTO 40. In a moment, we'll use a variation of this technique to allow for several overlays.

Setting Up Files

Let's write an example geared to the Commodore 64. We'll overlay three items: a graphics screen and two small machine language programs. The screen will load into the usual screen memory area, locations 1024–2023. The machine language programs will come into the cassette buffer, which starts at location 828 on the 64. (Because this example uses the cassette buffer, it works only with disk.) Only one machine language module will be in memory at a time.

Enter NEW, then type in this program. It creates a screen that will be loaded later.

```
100 DATA 8,1,16,16,25,32,2,
    9,18,20,8,4,1,25
110 OPEN 1,8,2,"0:SCREEN,P,
    W"
120 PRINT#1,CHR$(0);CHR$(4)
    ;
130 FOR J=1 TO 986
140 PRINT#1,CHR$(32);
150 IF J<>494 GOTO 200
160 FOR K=1 TO 14
170 READ X
180 PRINT#1,CHR$(X);
190 NEXT K
200 NEXT J
210 CLOSE 1
```

Make sure that lines 120, 140, and 180 each end with a semicolon. When you run this program, it creates a file called SCREEN which is four disk blocks in length. When that's done, enter NEW again and type in the next program. This one creates a machine language program called MLA. When the ML program loads into memory, it will do three small jobs: It will clear the screen, change the screen background color to white, and set the screen's POKE color to red.

```
100 DATA 60,3
110 DATA 169,147,32,210,255
120 DATA 169,31,32,210,255
130 DATA 169,1,141,33,208
140 DATA 169,0,133,252,169,
    216,133,253
150 DATA 162,4,169,2,160,0
160 DATA 145,252,200,208,25
    1
170 DATA 230,253,202,208,24
    6,96
200 A=42
210 FOR J=1 TO A
220 READ X
230 T=T+X
240 NEXT J
250 IF T<>6238 THEN STOP
260 RESTORE
270 OPEN 8,8,8,"0:MLA,P,W"
280 FOR J=1 TO A
290 READ X
300 PRINT#8,CHR$(X);
310 NEXT J
320 CLOSE 8
```

Be sure that line 300 ends with a semicolon. Run the program; if it stops at line 250, you have an error in one of the DATA statements.

Once that's done, enter NEW again. The next generator program creates a machine language routine to blink the screen. This ML module, which we'll call MLB, will occupy the same part of memory as MLA. The memory conflict isn't important since we'll load the programs one at a time. Type in and run this program:

```
100 DATA 60,3
110 DATA 169,0,133,252,173,
    136,2
120 DATA 133,253,162,4,160,
    0
130 DATA 177,252,201,32,240
    ,4
140 DATA 73,1?28,145,252,200
    ,208,243
150 DATA 230,253,202,208,23
    8,96
200 A=34
210 FOR J=1 TO A
220 READ X
230 T=T+X
240 NEXT J
250 IF T<>5022 THEN STOP
260 RESTORE
270 OPEN 8,8,8,"0:MLB,P,W"
280 FOR J=1 TO A
290 READ X
```

```
300 PRINT#8,CHR$(X);
310 NEXT J
320 CLOSE 8
```

Be sure to put a semicolon at the end of line 300. If you've typed the program correctly, it writes the ML program MLB to disk. At this point, all of the modules are complete. Let's write the main program to tie it all together.

The Main Program

Enter NEW and type in the following program lines. We'll start with a line that dispatches the program to the correct line after each load:

```
100 ON X GOTO 130,160,180
```

The first load brings in the machine language program MLA.

```
110 X=1
120 LOAD "0:MLA",8,1
```

After the first load is complete, line 100 sends us to line 130, where we activate the ML program with SYS:

```
130 SYS 828
```

The next two lines bring in the graphics screen.

```
140 X=2
150 LOAD "0:SCREEN",8,1
```

When the screen has loaded, you'll see the message it contains. After the second load is done, line 100 sends us to line 160, where we bring in the second machine language program:

```
160 X=3
170 LOAD "0:MLB",8,1
```

We resume at line 180 (courtesy of line 100) with a screen in place, the colors set as desired, and a blink program waiting to be called with another SYS command. Let's finish off with a loop to flash the message.

```
180 FOR J=1 TO 20
190 SYS 828
200 FOR K=1 TO 100
210 NEXT K
220 NEXT J
```

That's all it takes. It's a simple example, but the program shows the potential of the overlay technique.

Self-Chaining

Earlier in this series, we mentioned *self-chaining*, a method of restarting a program that has snarled itself inside several levels of subroutines. Again, keep in mind that prevention is the best way to avoid this problem. Good program structure should ensure that you never get tangled up in your own code. But occasionally you may program yourself into a corner and need a simple way to get out.

Assuming that you've gotten into this deplorable situation somehow, you can escape by making the dubious program chain to itself. The chaining activity cancels all FOR-NEXT loops and subroutine RETURNs, and also RESTOREs the DATA pointer to the very first DATA statement in the program. However, all existing variables are preserved, and all open files (if any) remain open.

Don't misunderstand what a self-chain does. The program text itself doesn't change—all you've done is reload the same program lines into memory. But the act of doing so untangled the snarled subroutines and FOR-NEXT loops and restarted the program from its first line. Other than that, everything remains as it was before the self-chain.

Since it's the chaining (not the loading) that does the trick, we can skip loading the program itself. Instead, we can overlay a single byte somewhere in memory to trigger the chaining process. To illustrate, let's write to disk a simple one-byte program file that will load the useless byte to some unimportant memory location. The chaining action that accompanies the load will do the job we want.

To write this file, type NEW and enter the following program:

```
100 DATA 255,0,0
270 OPEN 8,8,8,"0:DUMMY,P,W
    "
280 FOR J=1 TO 3
290 READ X
300 PRINT#8,CHR$(X);
310 NEXT J
320 CLOSE 8
```

Again, be sure that there is a semicolon at the end of line 300. When you run this program, it creates a tiny file named DUMMY. Now let's repeat the dreadful program that we used before. Again, please don't write programs this way; it's here just to illustrate the point. Type NEW and enter this program:

```
100 IF N>0 GOTO 130
110 PRINT "NAME LIST"
120 DIM N$(50)
130 PRINT
140 PRINT "DO YOU WANT TO -
    -"
150 PRINT "1. ENTER NAMES"
160 PRINT "2. LIST NAMES"
170 PRINT "3. QUIT"
180 INPUT "YOUR CHOICE";C
190 ON C GOSUB 210,310,350
200 GOTO 130
210 PRINT "ENTER EACH NAME"
220 PRINT "FOLLOWED BY AN '
    *' CHARACTER"
230 PRINT "TO END ENTRY"
240 GOSUB 260
250 GOTO 240
260 INPUT N$
270 IF N$="*" OR N=50 THEN
    LOAD "DUMMY",8
280 N=N+1
290 N$(N)=N$
300 RETURN
310 FOR J=1 TO N
320 PRINT N$(J)
330 NEXT J
340 RETURN
350 END
```

Try to write programs in such a way that you don't get into the problem shown above. By the time the program reaches line 210, it's in a subroutine. At line 260, it's nested within a second subroutine. When line 270 discovers that an exit is wanted, we're almost stuck and don't dare GOTO 130, which would leave unRETURNed subroutine addresses on the computer's internal stack.

Here's how to escape. At line 270, LOAD the one-byte DUMMY file. The load does nothing, but the act of chaining untangles the rest of the mess. How does this compare to our first solution of the same problem, where the entire program chained to itself? You get the same results, but gain in speed because you're loading a much smaller file.

Overlaying, like the other methods examined in this series, becomes especially useful in big-program situations, and generally eases the burden of bringing large amounts of data into memory when it's needed. The computer still thinks that it's performing a chain, but overlaying uses the same general technique for a different purpose. Once you understand the difference between chaining and overlaying, you can write even more powerful, flexible programs. ©

Custom Title Bars For ST BASIC

George Miller, Assistant Technical Editor

This short program demonstrates how to put a custom title on ST BASIC's Output window. It works on all Atari ST-series computers.

ST BASIC puts four windows on the screen entitled Command, List, Edit, and Output. The Output window is where your programs actually run, and the window always displays the same title at the top of the screen: Output. By now you're probably tired of staring at this title bar and wish there was some way to change it.

Fortunately, there is. Not with a built-in BASIC command, however. You have to call a routine in a part of the ST's operating system known as AES (Application Environment Services). The job is not difficult, but the ST BASIC manual lacks the necessary information for making system calls.

When programming the ST, it's helpful to remember that the operating system contains many routines which can be of help. These routines are part of GEM, the Graphics Environment Manager, which is divided into two sections: AES and the VDI (Virtual Device Interface). These libraries contain almost all the routines necessary to handle screen output. Although VDI and AES routines are most easily accessed by programmers using C or machine language, ST BASIC programmers can also call them with the VDISYS and GEMSYS commands. It requires a little extra effort, though.

The short routine listed below, "Custom Title Bars," is an example of a GEMSYS call to the AES library. It can be inserted into any ST BASIC program to display your program's title on the Output window's title bar. Run the routine to see what it does; then modify it in the following ways when using it in your own programs:

1. Change line 20 to assign to the string variable *title$* the name to be displayed in the title bar.

2. Delete line 40, the END statement, and insert your own program at this point. However, be sure you insert an END statement at the end of your program and before line 63000. Otherwise, your program will fall through into the subroutine and cause an error.

Before actually making the GEMSYS call in line 63040, the routine POKEs several parameters into system variables at the addresses pointed to by the built-in BASIC variable *gintin*. These parameters are required by this AES routine. The setup is done in lines 63010–63040.

More information about calling VDI and AES routines can be found in the Atari documentation available to software developers and in *COMPUTE!'s ST Programmer's Guide*, published by COMPUTE! Books.

Custom Title Bars

```
10 FULLW 2 : CLEARW 2
20 title$="New Title" : 'Define title$ =
   program title.
30 GOSUB titlebar
40 END : 'Start your program here.
63000 titlebar : 'Custom title bar routine.
63010 a# = gb : gintin = PEEK(a#+8)
63020 POKE gintin+0,PEEK(systab+8) :
      POKE gintin+2,2
63030 s# = gintin+4 : title$ = title$ +
      CHR$(0)
63040 POKE s#,varptr(title$) : GEMSYS
      (105)
63050 RETURN
```

Looking Glass: Windows For The 64

James E. Hosek

This interesting program adds two new commands to Commodore 64 BASIC which let you create text windows and pull-down menus similar to those on Commodore 128, Atari ST, Amiga, and Macintosh computers. You can also add four text screens of information, including help screens.

"Looking Glass" is an all machine language utility that brings advanced windowing capabilities to the Commodore 64. Since it works as an extension to BASIC, you can use this program without understanding machine language at all.

To get started, type in the data from Program 1 using the "MLX" machine language entry program published elsewhere in this issue. Here are the starting and ending addresses you need for MLX:

Starting address: C000
Ending address: C62F

When you're finished entering all the data, be sure to use the MLX Save option to save at least one copy. If you want to try out the examples detailed below, be sure to save the data with the filename LG.

To use Looking Glass, load it with LOAD"LG",8,1 for disk or LOAD"LG",1,1 for tape. Activate it by typing SYS 49152 and pressing RETURN.

You now have two new BASIC statements for creating windows and menus. The OPENW (Open Window) statement opens a window on the text screen from any of five different sources (see below). The SAVEW (Save Window) statement saves the contents of an existing window into one of the four available workspaces which Looking Glass uses.

The new BASIC statements work either in immediate mode (when you're not running a program) or in program mode. Just like normal BASIC keywords, they can be abbreviated if you wish. The abbreviation for OPENW is O SHIFT-P W. The abbreviation for SAVEW is S SHIFT-A W.

To use Looking Glass in a program of your own, include these lines:

```
10 IF PEEK(49152)<>169 THEN LOAD
   "LG",8,1
20 SYS 49152
```

If you're using tape instead of disk, change the ,8,1 to ,1,1.

OPENW Opens Windows

Here is the general format for the OPENW statement:

OPENW *s,x,y,w,h,f*

OPENW can use from one to six parameters (values). The first parameter (*s* in this example) can range from 0 to 9 and must always be present. This value tells Looking Glass the *source* of the text which will appear in the new window. A source value of 0 designates the normal text screen (memory locations 1024–2023) as the source for the window. Opening a window from source 0 does not change what's currently displayed, since it merely copies the current contents of screen memory into the same locations.

Source values 1–4 designate one of the four workspaces which Looking Glass allocates underneath the 64's Kernal ROM. As you'll learn below, these areas initially contain garbage; the SAVEW command can be used to store meaningful information there.

Source values 5–9 have a special function. They automatically create a window the same size as the entire screen, using one of the four workspaces as a source of information. When you specify a source from 5 to 9, only the first parameter is relevant; Looking Glass ignores all additional parameters (see below).

Window Coordinates

The second and third parameters in an OPENW command (indicated by *x* and *y* in the previous example)

locate the upper-left corner of the window you want to open. Specifying this corner's location effectively determines the screen position for the entire window. The horizontal (*x*) coordinate can range from 0 to 39, and the vertical (*y*) coordinate can range from 0 to 24.

The next two parameters (*w* and *h* in the previous example) represent the *width* and *height* of the window, respectively. The width value can range from 1 to 40, while the height value can range from 1 to 25. Note, however, that the maximum width and height for a given window depends on where its upper-left corner is located. For instance, if you locate the upper-left corner 10 columns from the left edge of the screen, you won't have room for a window that's 40 columns wide. To keep everything on the screen, you must make sure that a window's horizontal coordinate plus its width doesn't exceed 40, and that its vertical coordinate plus its height doesn't exceed 25.

The last parameter (*f* in the previous example) specifies the type of *frame* the window will have, and whether the window's contents will be normal or reverse video. A frame value of 0 creates a frameless window. A value of 2 selects a normal frame, and 4 creates a reversed frame. To make the window appear in reverse video, add 1 to any of the previous three values. The table below outlines the options for the frame parameter.

Table: Frame Parameter

0	No frame, normal window
1	No frame, reverse window
2	Normal frame, normal window
3	Normal frame, reverse window
4	Reverse frame, normal window
5	Reverse frame, reverse window

Any of the parameters for OPENW can be specified as a constant, variable, or arithmetic expression. For example, if S=1, then the statement OPENW S has the same effect as OPENW 1. If you omit a parameter, it defaults to the most recently used value (if any). To allow room for the frame, framed windows must have a width and height of at least three. Here are a few examples of legal OPENW commands:

```
OPENW 1,10,10,20,5
OPENW 4,,,25,10
OPENW 2,X,Y,10+X*2,5+Y*3,F
```

Saving With SAVEW

The SAVEW command saves the contents of a window in one of the special Looking Glass workspaces. This is useful when you need to save the contents of a window for further use and for certain other purposes which we'll explain below. Here is the general format for SAVEW, which takes only one parameter:

```
SAVEW w
```

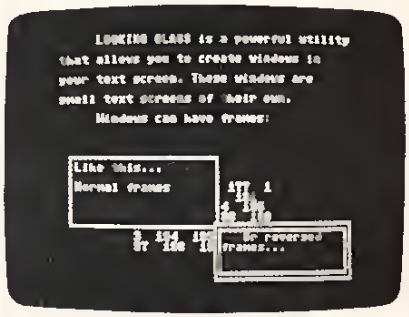
In this example *w* stands for *workspace*, and corresponds to the values used for the source in an OPENW command. Legal workspace values can range from 0 to 9. If you SAVEW with a value from 1 to 4, Looking Glass saves the contents of the current window in one of the four workspaces located under ROM. If you SAVEW with a value from 6 to 9, Looking Glass saves the entire display screen (which may be bigger than the current window) in the designated workspace.

Thus, after deciding which workspace to use, you have a basic choice between saving an entire screenful of information or saving only the contents of a window. Note that SAVEW stores the contents of a window or screen without disturbing what's already there. Values of 0 and 5 are legal for SAVEW, but have no visible effect since they simply store the contents of the current window or screen back into their present locations.

Working Inside Windows

After you open a window with a screen number of 0 to 4, certain restrictions apply. All text and output go only into the defined window area. Windows scroll separately from the rest of the screen, and a screen clear operation clears only the window. In immediate mode, commands can occupy only one physical line, without any wraparound at the window's edge. (If you wish to edit a program after creating a window, either press RUN/STOP–RESTORE or execute an OPENW command with a source value of 5 to 9.)

Windows also affect the behavior of the INST/DEL key and certain control codes for printing. When you type inside a window, either in direct mode or in response to an INPUT statement, the INST key (SHIFT–INST/DEL) always inserts a space at the cursor until the current line is full. DEL always deletes the character to the left of the cursor. If the cursor is at the beginning of a line, it wraps back to the end of the previous line, but does *not* pull any text with it. Looking Glass ignores CHR$(20) and CHR$(148) when they are printed to screens 0–4.



"Looking Glass" adds advanced windowing capabilities to Commodore 64 BASIC, making it easy to create and manipulate windows like this.

When you type inside a reversed window in immediate mode, control characters do not work when embedded in quotation marks. For example, typing PRINT "{HOME}" prints the letter *S* instead of homing the cursor as usual. To circumvent this problem, either type PRINT CHR$(147) or specify a nonreversed window. However, the control *keys* (RVS ON, CLR, BLK, and so on) work normally in every window.

When PRINTing inside a window, the SPC function works normally, since it refers to the current cursor position. The TAB function, however, refers to the left edge of the *screen*, not the left edge of the window, and may cause unexpected results unless the two edges coincide. Avoid using commas to separate items for printing (for instance, as in the statement PRINT X,Y,Z). When you separate printed items with commas, the computer arranges them into columns that are multiples of ten spaces—which may or may not fall inside the current window.

You will probably find the string functions (LEFT$, RIGHT$, MID$) and the semicolon (;) most useful for formatting text inside a window. If you exit a window by pressing RUN/STOP–RESTORE, don't forget to reactivate Looking Glass with SYS 49152 before trying to use OPENW or SAVEW again.

A Graphic Demonstration

Let's try some experiments to become familiar with windowing. First, activate Looking Glass as described above. Then clear the screen and enter the following statement in immediate mode (without a line number):

```
OPENW 1,5,5,30,15,2
```

A large boxful of random characters appears in the middle of the screen. Press SHIFT–CLR/HOME to get rid of the garbage characters. If you move around the window with the cursor keys, you'll notice that the window is actually only 28 × 15; the rest of the space is taken up by the frame. Enter a few direct commands to get a feel for how the window works. For instance, you may want to load a BASIC program, LIST it in the window, change the character colors, and so on.

Now type SAVEW 2 and press RETURN. This command stores the contents of the window in workspace 2. (Notice that you don't have to SAVEW a window to the same workspace that was used when you opened it.) Press SHIFT–CLR/HOME again, then enter OPENW 2. This retrieves the stored information from workspace 2. The frame color is the last color that you specified; all other window parameters default to their previous values.

To open a reversed window, enter this command:

```
OPENW 1,,,,1: PRINT CHR$(147)
```

Note that the window is now a full 30 × 15. PRINTing CHR$(147) clears the window immediately so that no garbage appears. If you still have a program in memory, LIST it to confirm that the text indeed PRINTs in reverse video. To change the text color, press CTRL and any color key, then press SHIFT–CLR/HOME. The entire window changes to the selected color.

Press CTRL–RVS ON and type a few characters. Characters that are actually normal now appear in reverse mode. Next, enter OPENW 5 to leave the window and enter full screen mode. If you press SHIFT–CLR/HOME at this point, the whole screen is cleared. Enter OPENW 7. The previously stored text is now instantly recalled, along with the garbage that was not previously overwritten.

More Hints

The following line can be used to clear all four workspaces at the beginning of a program:

```
30 PRINT CHR$(147): SAVEW 6: SAVEW
   7: SAVEW 8: SAVEW 9
```

In some cases, you'll want two windows to overlap, but also be able to restore either window at any time. To accomplish this, save each window to a different workspace as soon as it is complete (that is, as soon as you're done printing in it). To restore the window, open it again with OPENW, using the same workspace number used when you saved it.

Sometimes it may be desirable to put a header or title in the frame of a window. The following example opens a 15 × 15 window with a normal frame and the header DIRECTORY:

```
100 OPENW 1,10,5,15,15,2: PRINT
    CHR$(147)
110 OPENW 0,10,5,15,1,0: PRINT
    "[RIGHT]DIRECTORY";
120 OPENW 0,11,6,13,13
```

Notice that line 120 opens from window 0 and that *x* and *y* are incremented by one, and *w* and *h* are decremented by two. In this case the *f* parameter defaults to zero, preventing Looking Glass from redrawing the frame and erasing the header.

While Looking Glass does not use any of the 64's BASIC programming space, it does use virtually all the RAM underneath the Kernal ROM, as well as RAM from locations 49152–50728 ($C000–$C628). The 64's BASIC ROM is also copied to underlying RAM and modified.

The more you learn about how Looking Glass works, the more uses you'll find for it. A pull-down menu, for instance, is simply a window located on the top edge of the screen. Program 2 demonstrates how to create nondestructive pull-down menus as well as many other unique effects. Once you master the techniques involved, you'll probably think of even more applications.

Program 1: Looking Glass

For instructions on entering this listing, please refer to the "MLX" article published in this issue of COMPUTE!.

```
C000:A9 51 8D 08 03 A9 C3 8D B0
C008:26 03 A9 E6 8D 02 03 A9 25
C010:DD 8D 04 AC A9 F6 8D 18 8B
C018:03 A9 C5 8D 19 03 A9 C0 FF
C020:8D 09 03 A9 C3 8D 27 03 4B
C028:A9 C4 8D 03 03 A9 C5 8D 69
C030:05 AC A9 00 85 FB A9 A0 A4
C038:85 FC A0 00 B1 FB 91 FB 6C
C040:E6 F8 D0 F8 E6 FC A5 FC 51
C048:C9 C0 D0 F0 A9 76 85 01 3B
C050:60 20 73 00 8D 16 C6 C9 94
C058:9F D0 0E A0 01 B1 7A C9 37
C060:57 D0 18 A9 06 85 FB D0 6E
C068:18 C9 94 D0 0E A0 01 B1 AE
C070:7A C9 57 D0 06 A9 01 85 F7
C078:FB D0 06 20 79 00 4C E7 3B
C080:A7 20 73 00 20 73 00 88 A3
C088:98 48 20 9E AD A9 00 2A 94
C090:20 90 AD 20 1B BC 20 BF C9
C098:B1 68 A8 A5 65 99 01 C6 D6
C0A0:20 79 00 C9 2C D0 1C 20 2A
C0A8:73 00 C8 C9 2C D0 02 F0 33
C080:F6 C4 FB F0 0E D0 D1 AC 71
C0B8:09 C6 20 CA F1 88 C0 02 56
C0C0:D0 F8 60 AD 01 C6 30 0E 61
C0C8:C9 0A 10 0A AE 16 C6 E0 90
C0D0:94 D0 0D 4C 2A C3 A9 00 EA
C0D8:8D 19 C6 A2 0E 6C 00 03 8F
C0E0:AD 01 C6 C9 05 30 07 18 FD
C0E8:6E 19 C6 4C DE C2 38 6E 66
C0F0:19 C6 A0 04 B9 01 C6 99 FD
C0F8:06 C6 88 D0 F7 AD 07 C6 98
C100:18 6D 09 C6 C9 29 10 CE 5A
C108:AD 08 C6 18 6D 0A C6 C9 A9
C110:1A 10 C3 A9 00 8D 17 C6 E2
C118:AD 06 C6 4A 8D 18 C6 6E 3A
C120:17 C6 C9 03 10 B0 C9 00 21
C128:F0 02 A9 02 CD 09 C6 10 2A
C130:A5 CD 0A C6 10 A0 AD 18 1E
C138:C6 C9 00 D0 03 4C F6 C1 97
C140:AD 18 C6 C9 01 F0 04 A9 93
C148:12 D0 02 A9 92 20 CA F1 80
C150:AE 08 C6 AC 07 C6 18 86 DA
C158:02 20 F0 FF E6 02 A9 B0 46
C160:20 CA F1 A9 60 20 87 C0 33
C168:A9 AE 20 CA F1 AD 0A C6 3E
C170:38 E9 02 85 FB A6 02 AC 4E
C178:07 C6 18 20 F0 FF A9 7D 8E
C180:20 CA F1 A5 D3 18 6D 09 42
C188:C6 A8 88 84 D3 A9 7D 77
C190:20 CA F1 E6 02 C6 FB D0 77
C198:DC A6 02 AC 07 C6 18 20 E2
C1A0:F0 FF A9 AD 20 CA F1 A9 66
C1A8:60 20 B7 C0 AD 08 C6 18 9A
C1B0:6D 0A C6 C9 19 D0 23 AD E2
C1B8:07 C6 18 6D 09 C6 C9 28 6A
C1C0:D0 18 AD 18 C6 C9 02 F0 3C
C1C8:04 A9 7D D0 02 A9 FD 8D B5
C1D0:E7 07 AD 86 02 8D E7 DB 1A
C1D8:D0 05 A9 BD 20 CA F1 EE 16
C1E0:07 C6 EE 08 C6 CE 09 C6 42
C1E8:CE 09 C6 CE 0A C6 CE 0A EE
C1F0:C6 A9 00 8D 18 C6 A9 00 4A
C1F8:8D 21 C6 AC 01 C6 A9 04 A9
C200:8D 1F C6 B9 0C C6 8D 22 41
C208:C6 20 61 C2 AC 01 C6 A9 F1
C210:D8 8D 1F C6 B9 11 C6 8D E2
C218:22 C6 20 61 C2 A9 92 4D A9
C220:17 C6 20 CA F1 AE 08 C6 B4
C228:AC 07 C6 18 20 F0 FF AD 92
C230:07 C6 18 6D 09 C6 A8 8C 06
C238:1B C6 88 8C 1A C6 AD 08 26
```

```
C240:C6 18 6D 0A C6 A8 8C 1D 8C
C248:C6 88 8C 1C C6 AC 08 C6 66
C250:B9 D9 00 09 80 99 D9 00 D7
C258:C8 CC 1D C6 30 F2 4C DB 47
C260:C2 AC 0A C6 A9 00 85 FC 75
C268:AD 08 C6 0A 0A 18 6D 08 D3
C270:C6 0A 0A 26 FC 0A 26 FC D8
C278:18 6D 07 C6 85 FB 90 02 F1
C280:E6 FC 18 A5 FB 6D 21 C6 B4
C288:85 FD AD 22 C6 65 FC 85 73
C290:FE 18 AD 1F C6 65 FC 85 8E
C298:FC 78 A5 01 48 A9 75 85 D8
C2A0:01 98 AA AC 09 C6 88 B1 13
C2A8:FD 2C 1E C6 10 06 AD 23 7F
C2B0:C6 0D 17 C6 91 FB 88 10 C9
C2B8:EE 18 A9 28 65 FB 85 FB 95
C2C0:A9 00 65 FC 85 FC 18 A9 91
C2C8:28 65 FD 85 FD A9 00 65 CF
C2D0:FE 85 FE CA D0 CD 68 85 D7
C2D8:01 58 60 4C 7B C0 38 E9 FE
C2E0:05 8D 01 C6 A9 00 8D 19 5A
C2E8:C6 20 EF C2 4C F6 C1 A9 6F
C2F0:00 8D 07 C6 8D 08 C6 A9 EA
C2F8:28 8D 09 C6 A9 19 8D 0A 5A
C300:C6 60 A9 00 8D 21 C6 AC 63
C308:01 C6 A9 04 8D 22 C6 B9 73
C310:0C C6 8D 1F C6 20 61 C2 2F
C318:AC 01 C6 A9 D8 8D 22 C6 81
C320:89 11 C6 8D 1F C6 20 61 30
C328:C2 60 C9 05 10 06 20 02 8D
C330:C3 4C 7B C0 38 E9 05 8D 29
C338:01 C6 A2 03 BD 07 C6 48 56
C340:CA 10 F9 20 EF C2 20 02 3F
C348:C3 E8 68 9D 07 C6 E8 E0 D8
C350:05 D0 F7 4C 78 C0 8E 24 72
C358:C6 8C 25 C6 AC 0A C6 C0 53
C360:01 F0 25 A9 D8 8D 1F C6 E5
C368:8D 22 C6 A9 28 8D 21 C6 33
C370:88 20 64 C2 A9 04 8D 1F 94
C378:C6 8D 22 C6 A9 28 8D 21 A1
C380:C6 AC 0A C6 88 20 64 C2 94
C388:38 6E 1E C6 A9 20 8D 23 04
C390:C6 A9 00 85 FC AD 1C C6 DB
C398:0A 0A 6D 1C C6 0A 26 FC BE
C3A0:0A 26 FC 0A 26 FC 18 6D B9
C3A8:07 C6 85 FB A9 00 69 04 F9
C3B0:65 FC 85 FC A0 01 20 99 8D
C3B8:C2 0E 1E C6 AE 24 C6 AC 95
C3C0:25 C6 60 08 2C 19 C6 30 9C
C3C8:04 28 4C CA F1 85 02 8E CA
C3D0:26 C6 8C 27 C6 38 20 F0 69
C3D8:FF A5 02 C9 0D D0 20 A9 3C
C3E0:00 85 D4 2C 17 C6 10 02 1D
C3E8:A9 12 85 C7 EC 1C C6 30 8C
C3F0:08 20 56 C3 AE 1C C6 10 0F
C3F8:01 E8 AC 07 C6 10 40 C9 02
C400:8D F0 DC 48 A5 D4 F0 04 13
C408:68 4C B8 C4 68 C9 93 D0 9E
C410:24 38 6E 1E C6 A9 20 8D 14
C418:23 C6 A9 04 8D 1F C6 20 F0
C420:61 C2 AD B6 02 8D 23 C6 7C
C428:A9 D8 8D 1F C6 20 61 C2 9C
C430:4E 1E C6 10 04 C9 13 D0 80
C438:16 AC 07 C6 AE 08 C6 18 80
C440:20 F0 FF AE 26 C6 AC 27 CD
C448:C6 A5 02 28 58 18 60 C9 0F
C450:91 D0 07 EC 08 C6 F0 EB AF
C458:D0 19 C9 11 D0 0C EC 1C 87
C460:C6 10 02 30 0E 20 56 C3 F5
C468:B0 D9 C9 1D D0 1F CC 1A 82
C470:C6 10 0A 28 AE 26 C6 AC 6D
C478:27 C6 4C CA F1 EC 1C C6 BF
C480:30 05 20 56 C3 B0 01 E8 98
C488:AC 07 C6 10 B2 C9 9D D0 CC
C490:12 CC 07 C6 F0 02 D0 DB B0
C498:EC 08 C6 F0 A6 AC 1A C6 65
C4A0:CA 10 9C C9 14 F0 9C C9 2B
C4A8:94 F0 98 C9 12 D0 05 4D 93
C4B0:17 C6 D0 BF C9 92 F0 F7 FF
C4B8:CC 1A C6 F0 02 D0 B4 EC C0
C4C0:1C C6 F0 10 20 CA F1 E8 22
C4C8:B5 D9 09 80 95 D9 AC 07 41
C4D0:C6 4C 3F C4 EC 08 C6 F0 0B
C4D8:EF 20 56 C3 CA 18 20 F0 51
C4E0:FF A5 02 4C C4 C4 2C 19 83
C4E8:C6 30 03 4C 83 A4 20 F4 EA
C4F0:C4 4C 86 A4 A4 D3 81 D1 B4
C4F8:85 FE A9 00 85 CC 20 E4 BE
C500:FF AA F0 F6 48 A9 01 85 34
C508:CC A5 FE A4 D3 91 D1 AD C3
C510:87 02 91 F3 68 C9 0D F0 C6
C518:11 C9 8D F0 0D C9 14 F0 08
C520:6F C9 94 F0 37 20 D2 FF 57
C528:90 CA AC 07 C6 AE 09 C6 7E
C530:A9 00 85 FB A9 02 85 FC 5E
C538:88 E8 B1 D1 8C 27 C6 4D 71
C540:17 C6 A8 30 19 C9 20 10 61
C548:05 18 69 40 90 16 C9 40 3E
C550:10 02 30 10 C9 60 10 F1 4D
C558:09 80 D0 08 F0 5D 29 7F F1
C560:C9 40 10 E5 A0 00 91 FB 65
C568:AC 27 C6 E6 FB C8 CA D0 C4
C570:C9 A0 00 98 91 FB C6 FB 98
C578:B1 FB 29 3F C9 20 D0 06 6B
C580:A9 00 91 F8 F0 F0 A9 0D 7E
C588:20 D2 FF A2 00 A0 02 60 E9
C590:A4 D3 CC 07 C6 F0 14 81 41
C598:D1 88 91 D1 C8 C8 CC 1B 9C
C5A0:C6 D0 F4 88 A9 20 0D 17 E9
C5A8:C6 91 D1 A6 D4 A9 00 85 73
C5B0:D4 A9 9D 20 D2 FF 86 D4 3F
C5B8:4C F4 C4 AC 1A C6 B1 D1 2C
C5C0:29 7F C9 20 D0 12 88 B1 8D
C5C8:D1 C8 91 D1 88 C4 D3 D0 8E
C5D0:F5 A9 20 20 D2 FF 90 D3 53
C5D8:A9 94 4C 25 C5 2C 19 C6 12
C5E0:30 03 4C 60 A5 38 20 F0 14
C5E8:FF 98 18 ED 07 C6 85 FA D5
C5F0:20 F4 C4 A6 FA 60 24 91 FF
C5F8:30 04 18 6E 19 C6 4C 47 4B
C600:FE 00 00 00 28 19 00 00 82
C608:00 00 00 00 04 E0 E8 F0 F8
C610:F8 D8 E4 EC F4 FC 00 00 57
C618:00 00 00 00 00 00 00 00 A5
C620:00 00 00 00 00 00 00 00 AD
C628:00 00 00 00 00 00 00 00 B5
```

Program 2: Window Demonstration

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" published in this issue of COMPUTE!

```
MC 10 IFPEEK(49152)<>169THENLO
      AD"LG",8,1
SQ 20 SYS49152
JK 30 POKE53280,11:POKE53281,1
      1:PRINT"{8}{CLR}";CHR$(1
      4)
QS 40 OPENW5:SAVEW6:SAVEW7:SAV
      EW8:SAVEW9
XP 50 PRINT"{CLR}{OFF}
      {5 SPACES}LOOKING
      {SHIFT-SPACE}GLASS IS A
      {SPACE}POWERFUL UTILITY"
FP 60 PRINT"THAT ALLOWS YOU TO
       CREATE WINDOWS IN":PRIN
      T
HB 70 PRINT"YOUR TEXT SCREEN.
      {SPACE}THESE WINDOWS ARE
      ":PRINT
JC 80 PRINT"SMALL TEXT SCREENS
       OF THEIR OWN.":GOSUB740
EX 90 PRINT"{WHT}";:OPENW1,10,
      15,19,7,0:PRINT"{CLR}";:
      FORX=1TO200:PRINTX;:NEXT
SB 100 SAVEW1: OPENW5:PRINT"
      {HOME}{CYN}{8 DOWN}
      {5 SPACES}WINDOWS CAN H
      AVE FRAMES:":GOSU8740
SC 110 PRINT"{6}";:OPENW2,1,12
      ,21,8,2:PRINT"{CLR}LIKE
       THIS..."
DK 120 PRINT"{DOWN}NORMAL FRAM
      ES{YEL}":SAVEW2:OPENW2,
      21,19,19,6,4
PP 130 PRINT"{CLR}{3 SPACES}OR
       REVERSED{3 SPACES}FRAM
      ES...":SAVEW2:GOSUB740:
      OPENW5
FH 140 PRINT"{WHT}{HOME}
      {8 DOWN}{5 SPACES}PLUS
      {SPACE}WINDOWS CAN BE R
      EVERSED:{CYN}":GOSUB740
MQ 150 OPENW3,5,11,15,14,5:PRI
      NT"{CLR}LIKE THIS...":G
      OSU8740
PD 160 FORX=1TO100:PRINTX;:NEX
      T:PRINT:SAVEW3:GOSU8740
      :OPENW4,0,0,40,10,0:PRI
      NT"{CLR}"
SF 170 PRINT"{7}{OFF}
      {5 SPACES}EVEN THOUGH T
      HE WINDOWS OVERLAP,":PR
      INT
KD 180 PRINT"THEY ARE NON DEST
      RUCTIVE:":GOSU8740
BD 190 OPENW1,10,15,19,7,0:GOS
      U8740:PRINT"{6}"
CR 200 OPENW2,1,12,21,8,2:GOSU
      B740:PRINT"{YEL}"
PG 210 OPENW2,21,19,19,6,4:GOS
      U8740:PRINT"{CYN}";:OPE
      NW3,5,11,15,14,5:GOSU87
      40
QH 220 OPENW5:SAVEW5:OPENW1,0,
      0,40,25,5:PRINT"{CLR}"
EQ 230 PRINT"{DOWN}{5 RIGHT}ON
      E OF THE POSSIBLE USES
      {SPACE}OF{DOWN}"
EP 240 PRINT" LOOKING GLASS IS
       TO CREATE PULL DOWN
      {DOWN}"
HX 250 PRINT" MENUS:"
FR 260 PRINT"{DOWN}{3 SPACES}
      {DOWN}TYPE P, T, OR F T
      O PULL DOWN MENU"
BM 270 PRINT"{DOWN}{3 SPACES}P
      RESS {RVS}RETURN{OFF} T
      O GO ON:":FR=0:PR=0:TV=
      0
EG 280 OPENW0,0,0,40,1,1:PRINT
      "{2 RIGHT}PRESIDENTS
      {4 RIGHT}TV SHOWS
      {6 RIGHT}FRUITS";
CC 290 OPENW5:SAVEW7
XC 300 X=PEEK(203):IFX=64THEN3
      00
GS 310 IFX=1THEN560
BM 320 IFX=41THEN360
PE 330 IFX=22THEN430
FK 340 IFX=21THEN480
FH 350 GOTO300
HF 360 PRINT"{GRN}":OPENW4,1,0
      ,12,17,5:IFPRTHEN410
PK 370 PRINT"{CLR}PRESIDENTS"
BQ 380 PRINT"WASHINGTONLINCOLN
      ":PRINT"ROOSEVELT":PRIN
      T"NIXON":PRINT"JOHNSON"
SH 390 PRINT"JEFFERSON":PRINT"
      FORD":PRINT"CARTER":PRI
      NT"REAGAN"
GX 400 PRINT"ADAMS":PRINT"MADI
      SON":PRINT"GRANT"
JS 410 X=PEEK(203):IFX=41THEN4
      10
BK 420 PR=-1:SAVEW4:OPENW7:GOT
      O300
GK 430 PRINT"{YEL}":OPENW4,14,
      0,12,12,5:IFTVTHEN460
8A 440 PRINT"{CLR} TV
      {SHIFT-SPACE}SHOWS
      {DOWN}":PRINT"A TEAM":P
      RINT"COSBY SHOW";
GM 450 PRINT"NIGHT{DOWN}
```

```
        {3 LEFT}COURT":PRINT"FA
        MILY{DOWN}{4 LEFT}TIES"
        :PRINT"LATE NIGHT";
XF 460  X=PEEK(203):IFX=22THEN4
        60
GD 470  TV=-1:SAVEW4:OPENW7:GOT
        O300
KJ 480  PRINT"{7}":OPENW4,27,0,
        12,23,5:IFFRTHEN540
PH 490  PRINT"{CLR}{2 SPACES}FR
        UITS"
JD 500  PRINT"{DOWN}APPLES":PRI
        NT"ORANGES":PRINT"BANAN
        AS":PRINT"PEARS":PRINT"
        LEMONS"
HQ 510  PRINT"KUMQUATS":PRINT"K
        IWI FRUITWATERMELONGRAP
        EFRUITTANGERINE STRAWBE
        RRY";
CQ 520  PRINT"PLUM":PRINT"PEACH
        ":PRINT"BLUEBERRY":PRIN
        T"APRICOT":PRINT"RASPBE
        RRY"
KF 530  PRINT"PINEAPPLE CHERRY"
        ;
DD 540  X=PEEK(203):IFX=21THEN5
        40
HD 550  FR=-1:SAVEW4:OPENW7:GOT
        O300
KX 560  PRINT"{CLR}{10 DOWN}
        {3 RIGHT}{RVS}PLEASE PU
        T ON YOUR SAFETY GOGGLE
        S"
DG 570  PRINT"{DOWN}{3 RIGHT}
        {7 SPACES}THEN PRESS RE
        TURN:":POKE198,0
QG 580  GETA$:IFA$<>CHR$(13)THE
        N580
ES 590  POKE198,0:PRINT"{CLR}":
        C$(0)="{CYN}":C$(1)="
        {YEL}":C$(2)="{GRN}":C$
        (3)="{WHT}"
SE 600  FORX=0TO70:PRINT"LOOKIN
        G GLASS "C$(XAND3);:NEX
        T:SAVEW6
DR 610  PRINT"{CLR}{RVS}":FORX=
        1TO23:PRINT"{39 +}":NEX
        T
JG 620  OPENW0,0,0,40,25,4
PP 630  OPENW0,6,5,29,15,0:PRIN
        T"{WHT}{CLR}":FORX=1TO1
        3:PRINT"{28 *}"
QX 640  NEXT:OPENW0,5,5,30,15,4
        :PRINT"{YEL}";:SAVEW7
GA 650  FORX=1TO10:OPENW1,19-X,
        12-X,2*X+2,2*X+2,4:GOSU
        B750:NEXT
XM 660  FORX=0TO9:OPENW2,9+X,2,
        1,22,0:OPENW1,10+X,2,21
        -X,22,4:GOSUB750:NEXT
MG 670  FORX=0TO9:OPENW2,18,2+X
        ,13,1,0:OPENW1,19,3+X,1
        2,21-X,4:GOSUB750:NEXT
PD 680  FORX=1TO10:OPENW2,20-X,
        24-X,12,1,0:OPENW2,31-X
        ,13-X,1,11,0
JM 690  OPENW1,19-X,12-X,12,12,
        4:GOSUB750:NEXT:SAVEW7
FM 700  FORX=1TO11:OPENW7:GOSUB
        750:GOSUB750:OPENW6:GOS
        UB750:GOSUB750:NEXT
BB 710  FORX=1TO50:PRINT"{DOWN}
        ";:NEXT
QJ 720  PRINT"PLEASE REMOVE YOU
        R SAFETY GOGGLES NOW.
        {10 DOWN}"
BD 730  END
QC 740  FORX=1TO1500:NEXT:RETUR
        N
BR 750  FORZ=1TO100:NEXT:RETURN
```


ST Hints & Tips

George Miller, Assistant Technical Editor

Here are some interesting tricks for setting up autobooting programs, customizing your GEM desktop, reading a joystick from ST BASIC, and souping up BASIC's performance with machine language subroutines. All the techniques work on the 520ST and 1040ST.

The Atari ST series computers are extremely powerful and complex machines. The numerous demo programs which are widely available offer only small peeks at the true capabilities of these computers. For programmers, however, the ST's power can be frustrating because it's so elusive. Virtually no technical documentation is supplied with the ST, and the two languages it comes with—Logo and ST BASIC—have their shortcomings.

If you invest $300 for an Atari development system package, you receive an assembler, a C compiler, and a huge mass of documentation on the Graphics Environment Manager (GEM), but most of it is not even ST-specific—it refers to GEM as implemented on the IBM PC.

However, careful study of this mountain of paper can reveal quite a few "secrets" about the ST. We'll let you in on a few of these tricks which enhance the power of your computer.

Autobooting Programs

How you ever wished that a certain program—perhaps a RAM disk utility, or an application, or a language—could run automatically when you start up your ST? This feature would be especially handy if you need to set up a disk for someone who wants to run a program without understanding anything more than how to turn on the computer.

The eight-bit Atari computers can automatically load and run programs by using AUTORUN.SYS files. Apple has the HELLO program, PC-DOS and AmigaDOS have batch files, and the Commodore 128 has provision for autobooting. Although it's not documented, so does your ST. Clues on how to create an autoexecute file can be found in GEMDOS.

As part of the initialization sequence, the ST looks for a folder called AUTO on the boot disk. Any files with a .PRG extender found in the AUTO folder are executed in sequence. These files are known as COMMAND.PRG files.

It's very easy to set up an autoboot program. Place your boot disk

in your drive, then point to the File heading on the menu bar. Select the New Folder option and create a folder named AUTO.

Move any program you want to autoboot into this folder. Any time you boot your ST from this disk, the program you placed in the AUTO folder will automatically run. This technique works with TOS in ROM or with the earlier disk-loaded TOS. There may be a problem, however, with autobooting some programs when using the high-resolution monochrome mode. Otherwise, it's the most foolproof autorun system yet.

Customizing The Desktop

Have you ever tried renaming your disk icons using the Install Drive option from the Options menu? Some characters can't be used. For instance, it isn't possible to name an icon Disk A because lowercase letters and spaces are not permitted. Also, you can't do anything with the trash can.

However, there is a way to change the names to anything you want. After saving your desktop, you can edit the file which stores the information for these options—DESKTOP.INF. For now we'll only change the icon names using this technique. Be careful to not change any other characters in the file.

First, you'll need a text editor such as *Mince* or *EMACS*, or even a word processor, like *ST Writer*. If you're using a word processor, set the left and top margins to zero.

The job itself is rather easy. Load the file DESKTOP.INF. It should look something like Figure 1.

Figure 1: DESKTOP.INF

```
#a000000
#b001100
#c77700070007000700552005055522207
  705570750577055041123O6
#d
#E 9B 03
#W 00 00 0C 01 1D 16 08 A:\*.*@
#W 00 00 28 01 1F 17 00 @
#W 00 00 0E 09 2A 0B 00 @
#W 00 00 0F 0A 2A 0B 00 @
#M 00 02 00 FF A FLOPPY DISK@ @
#M 00 03 00 FF B FLOPPY DISK@ @
#T 00 07 02 FF TRASH CAN@ @
#F FF 04 @ *.*@
#D FF 01 @ *.*@
#G 03 FF *.PRG@ @
#F 03 04 *.TOS@ @
#P 03 04 *.TTP@ @
```

Each character in this file is information about your desktop. Any change will affect what you see on the desktop and even how your ST functions to a certain extent. Use caution, since some changes might not yield the results you expect. To be safe, make sure you're working with a backup copy of your boot disk. Store the original in a safe location. This is always a good idea when experimenting with any file on a disk, and especially when modifying files that control the operation of your ST.

Now, move the cursor to the first line which begins with #M. Change the text, replacing the words FLOPPY DISK, so the line reads like this:

```
#M 00 02 00 FF A Disk A@ @
```

Then change the next line to:

```
#M 00 02 00 FF B Disk B@ @
```

If you want, you may change the name of the trash can icon. I called mine *Black Hole!* as a constant reminder that unlike the Amiga or Macintosh, the ST trash can does not let you easily recover files which are deleted. (There are some disk utilities available which allow you to recover trashed files, under limited conditions.)

To change the trash can icon, modify the next line to read:

```
#T 00 07 02 FF BIack Hole!@ @
```

The revised DESKTOP.INF file should be similar to Figure 2.

Figure 2: Revised DESKTOP.INF

```
#a000000
#b001100
#c77700070007000700552005055522207
  705570750577055041123O6
#d
#E 9B 03
#W 00 00 0C 01 1D 16 08 A:\*.*@
#W 00 00 28 01 1F 17 00 @
#W 00 00 0E 09 2A 0B 00 @
#W 00 00 0F 0A 2A 0B 00 @
#M 00 02 00 FF A Disk A@ @
#M 00 03 00 FF B Disk B@ @
#T 00 07 02 FF Black Hole!@ @
#F FF 04 @ *.*@
#D FF 01 @ *.*@
#G 03 FF *.PRG@ @
#F 03 04 *.TOS@ @
#P 03 04 *.TTP@ @
```

Finally, save the file back to the disk as DESKTOP.INF. The file must be saved in ASCII format, so make sure your text editor or word processor has this feature. If you're using *ST Writer* or some other word processors, it may be necessary to print the file to the disk in order to save it in ASCII format.

Reading The ST Joystick

ST BASIC is a fairly generic BASIC that has very few ST-specific commands. One of the most noticeably missing commands when you're trying to write a game is a function for reading the joystick. The ST works with any of the joysticks sold for the eight-bit Atari and Commodore computers, but there's no STICK or STRIG functions as in eight-bit Atari BASIC.

Actually, a joystick command does exist in the ST, but it's hidden deep within GEMDOS in the BIOS (Basic Input/Output System). This is an area not readily available from ST BASIC without using a few special techniques.

One easy way to find out what the joystick is doing is to ask the Intelligent Keyboard Device (IKBD). The keyboard has its own microprocessor, a 6301 chip, which is a member of the 6800 family. The keyboard processor is really a small computer system, with input/output lines, RAM, ROM, and even a serial interface which handles traffic to and from the 68000 central processing unit. The 68000 is not responsible for polling the keyboard continuously for activity. The 6301 notifies the 68000 via an interrupt when anything needs processing. In addition to reading the keyboard, the 6301 also reads the mouse, the joystick, and performs other functions.

The ST's link to the keyboard processor is through a chip called an ACIA (Asynchronous Communications Interface Adapter). The control register for the keyboard ACIA is located at memory address $FFFC00 in the ST, and the data register is at location $FFFC02. If you've moved to the ST from an earlier eight-bit computer, those may be the biggest hexadecimal numbers you've ever seen. Remember that the 68000 microprocessor in the ST has 24 address lines, enough for over 16,000,000 bytes of memory, as compared to the 65,536-byte maximum for eariler computers with only 16 address lines. For the ST you must become

accustomed to seeing hexadecimal addresses that are six digits long.

The following program is a short ST BASIC routine to read the values of the joystick plugged into port 1 (the rear joystick connector).

```
70 POKE &hfffc02,&h0012 'turn off
   mouse
80 POKE &hfffc02,&h0014 : joystick =
   PEEK(&hfffc02)
90 IF joystick = 511 THEN ? "north"
100 IF joystick = 2559 THEN ?
   "northeast"
110 IF joystick = 2303 THEN ? "east"
120 IF joystick = 2815 THEN ?
   "southeast"
130 IF joystick = 767 THEN ? "south"
140 IF joystick = 1791 THEN ?
   "southwest"
150 IF joystick = 1279 THEN ? "west"
160 IF joystick = 1535 THEN ?
   "northwest"
170 IF joystick < 0 THEN ? "fire button"
180 POKE &hfffc02,&h0008 'turn on
   mouse
190 GOTO 70
```

Line 70 sends a command to the IKBD, via the data register at $FFFC02, instructing it to turn off the mouse. (Note that ST BASIC uses &h to indicate hexadecimal numbers.)

Line 80 sends a command via the same address to turn on the joystick. Every movement of the joystick is reported to the processor. The joystick position is read by PEEKing the value returned in $FFFC02.

Lines 90–170 interpret the values returned from the IKBD.

Line 180 turns the mouse back on again. This should be done before exiting the program so the user will have control of the mouse when returning to BASIC or the desktop.

Line 190 makes the routine an infinite loop, so you'll need to press CTRL-C to stop this demonstration. If the mouse pointer isn't visible on the screen when the program stops, enter the following line and press RETURN to make the pointer reappear:

```
POKE &hfffc02,&h0008
```

To adapt this routine for use in your own programs, replace line 190 with 190 RETURN, then use GOSUB 70 to call the routine. Replace the PRINT statements in lines 90–170 with statements to perform the desired actions when the joystick is pressed in the indicated direction.

Mixing BASIC And Machine Language

To add real speed and power to any BASIC, it's often necessary to use machine language routines for certain tasks. In ST BASIC, machine language routines can be run using the CALL statement. The syntax for CALL is:

CALL *address variable,parameter list*

The *address variable* is a variable which holds the memory address of the beginning of the machine language routine. This location may be the address where the routine was loaded using the BLOAD command, or the address where the ML routine was POKEd. The *parameter list* is a list of values which can be passed to the ML routine. Some routines don't require any values to be passed, so this is optional.

The program below demonstrates how to POKE an ML routine into a variable, then use the VARPTR function to find the address to CALL.

As your library of ML routines expands, you'll find this method useful. Although the example program does nothing but print the letter A on the left side of the menu bar, it does demonstrate that ML routines give you full access to the ST, since the menu bar is usually off-limits to BASIC.

```
110 CLEARW 2 : FULLW 2
120 GOSUB init
130 ' ML opcodes in DATA statements
140 DATA &h3f3c,&h0041,&h3f3c,
   &h0002,&h4e41,&h588f
150 DATA &h3f3c,&h000d,&h3f3c,
   &h0002,&h4e41,&h588f
160 DATA &h3f3c,&h000a,&h3f3c,
   &h0002,&h4e41,&h588f,&h4e75
170 FOR i = 1 TO 19 : READ a : POKE
   x+(i*2),a : NEXT : 'POKE ml into
   ml$
180 CALL x
190 END
200 init : ml$="This is a dummy
   variable."
210 x = VARPTR (ml$)
220 RETURN
```

These tricks demonstrate only a small part of the ST's potential. Carefully studying the documentation reveals that some extremely powerful programming techniques are lurking just below the surface. If you're a curious programmer, explore GEM for ways to use the ST's features from within the tight BASIC framework. ©

Minding Memory From BASIC

D. W. Neuendorf

Are your programs fighting wars with each other for control of memory? Would you like to find a safe, protected place in RAM for machine language subroutines and other data in your BASIC programs? Here's how to use the memory management functions of PC-DOS to avoid conflicts and maximize the amount of memory available to BASIC. For the IBM PC, PCjr, and compatibles with DOS 2.0 or higher.

Over the past year, memory management in PC-DOS has become an important issue. The new desktop tools and coresident programs are designed to wait in the background to be called during the operation of another program. A number of these utilities may be lurking in memory at once, and programmers can't predict which other programs will be present with their own. The result can be memory conflicts and system crashes.

The designers of PC-DOS anticipated this situation to some extent. DOS 2.0 and later versions contain several function calls designed to give the operating system control over how the computer's memory is divided among programs residing in memory simultaneously. The most basic of these functions simply attempt to allocate and deallocate blocks of memory at a program's request. These DOS calls are readily available to machine language programmers, just like all other machine-level resources.

BASIC programmers, on the other hand, have no direct access to many DOS functions. But as we'll see, there are ways for BASIC programs to call on DOS to perform these memory management tasks.

Translating ML To BASIC

There are two DOS functions we're interested in—one for allocating memory and another for deallocating memory.

In machine language, both functions are called by placing a *function number* in the microprocessor's AH register and calling interrupt 21h. (Function numbers indicate to DOS which function is being called. The interrupt then performs the function.) The numbers are 48h for the allocate function and 49h for the deallocate function.

In addition to these numbers, each function call requires that you pass an argument. The allocate function requires the number of 16-byte *paragraphs* of memory to be allocated. This number must be placed in the microprocessor's BX register. The deallocate function requires the segment address of a block to be deallocated. This number must be placed in the ES register.

After each function is performed, it returns a value. The allocate function returns, via the AX register, either the segment address of an allocated block or an error code (7 or 8 plus a set carry bit) if the function was unsuccessful. The deallocation routine returns nothing if successful, but sets the carry bit and returns an error code (7 or 9) if unsuccessful. For those who are interested, Programs 1 and 2 show the assembler code necessary to call these functions.

Program 3 shows how to call these functions from BASIC. Since the allocate routine is not available initially and therefore can't allocate space for itself, the program reserves a few bytes for it just above BASIC (using the CLEAR statement in line 10). Once the allocate routine has been installed (lines 40–60), it can be used to get memory from DOS for machine language routines and other data. An example of its use is the call in line 70, which gets the segment address of a memory block for the deallocate routine.

Finally, line 120 shows an example of using the deallocate routine—it deallocates its own memory.

Program 1: DOS Memory Allocation

Note: This source code listing is for illustrative purposes only. It requires an assembler to enter.

```
                        page 50,132
0000                    alloc segment para
                            assume cs:alloc
                            assume ds:alloc
                            assume es:alloc
0000                    allocate proc far
                        ;
                        ;Routine to allow BASIC to make DOS
                         call to allocate a block of memory
                        ;outside of BASIC's own segment. CALL
                         ALLOC(MEMORY)-when BASIC calls the
                        ;routine, MEMORY contains the number
                         of bytes to be allocated.  When the
                        ;routine returns to BASIC, MEMORY
                         contains the segment address of the
                        ;allocated block of memory.  A 7 or 8
                         indicates allocation failed.
                        ;
0000  55                push bp
0001  8B EC             mov bp,sp
0003  8B 5E 06          mov bx,[bp+6]  ;get address of MEMORY
0006  8B 1F             mov bx,[bx]  ;get number of bytes to
                         be allocated
0008  B4 4B             mov ah,4Bh  ;DOS function number
000A  CD 21             int 21h  ;DOS call itself
000C  8B 5E 06          mov bx,[bp+6]  ;address of MEMORY
000F  89 07             mov [bx],ax  ;put segment address of
                         allocated memory in MEMORY
0011  5D                pop bp
0012  CA 0002           ret 2
                        ;
0015                    allocate endp
0015                    alloc ends
                        end
```

Program 2: DOS Memory Deallocation

Note: This source code listing is for illustrative purposes only. It requires an assembler to enter.

```
                        page 50,132
0000                    dealloc segment para
                            assume cs:dealloc
                            assume ds:dealloc
                            assume es:dealloc
0000                    dlc proc far
                        ;
                        ;Routine to allow BASIC to make DOS
                         call to deallocate a block of memory
                        ;Previously allocated using ALLOC. CALL
                         DEALLOC(MEMORY)-when BASIC calls the
                        ;routine, MEMORY contains the segment
                         address of the block of memory to be
                        ;dealloc.  When the routine returns to
                         BASIC, MEMORY contains either the
                        ;original segment address or an error
                         code.  A 7 or 9 indicates allocation
                         failed.
                        ;
0000  55                push bp
0001  06                push es
0002  8B EC             mov bp,sp
0004  8B 5E 06          mov bx,[bp+6]  ;get address of MEMORY
0007  8E 07             mov es,[bx]  ;get segment address of
                         block to be deallocated
0009  B4 49             mov ah,49h  ;DOS function number
000B  CD 21             int 21h  ;DOS call itself
000D  8B 5E 06          mov bx,[bp+6]
0010  89 07             mov [bx],ax  ;put error code in MEMORY
0012  07                pop es
0013  5D                pop bp
0014  CA 0002           ret 2
                        ;
0017                    dlc endp
0017                    dealloc ends
                        end
```

The Honor System

After studying Program 3, perhaps you've noticed another good reason for BASIC programmers to have access to these DOS calls: It's possible to put a machine language subroutine outside BASIC's 64K memory area, thus saving some space for BASIC programs. Better yet, you don't have to worry about where in memory you're hiding the routine—DOS takes care of it. If you use a lot of machine language subroutines, or store large amounts of data in memory, you'll have a lot more room to work with if you don't have to put everything inside BASIC's own segment.

One final comment about the DOS memory allocation functions: Please use them. Think of it as an honor system. If everyone relies on DOS to determine where their programs reside in memory, we can all feel confident that our coresident programs are not overlapping and conflicting with each other. But if too many programmers bypass these DOS functions, the rest of us won't dare to rely on them, either. After all, DOS can protect only the data or programs that it knows about.

Program 3: DOS Memory Functions in BASIC

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
IL 10 CLEAR ,&HFFDF:REM *** Rese
      rve a few bytes just above
       BASIC for alloc. routine
CL 20 DEFINT A-Z
JO 30 DEF SEG:ALLOC=&HFFDF:DMEMO
      RY=2:DEALLOC=0
KO 40 RESTORE 50:FOR X=0 TO 20:R
      EAD Y:POKE X+ALLOC,Y:NEXT:
      REM *** Install alloc.
IJ 50 DATA &h55,&hBb,&hec,&hBb,&
      h5e,&h06,&hBb,&h1f,&hb4,&h
      4B,&hcd
ED 60 DATA &h21,&hBb,&h5e,&h06,&
      hB9,&h07,&h5d,&hca,&h02,&h
      0
OJ 70 CALL ALLOC(DMEMORY):REM **
      * DOS call to allocate mem
      ory for dealloc. routine
FF B0 DEF SEG=DMEMORY
KH 90 RESTORE 100:FOR X=0 TO 22:
      READ Y:POKE X,Y:NEXT:REM *
      ** Install dealloc.
DE 100 DATA &h55,&h06,&hBb,&hec,
       &hB5,&h5e,&h06,&hBe,&h07,
       &hb4,&h49,&hcd
HN 110 DATA &h21,&hBb,&h5e,&h06,
       &hB9,&h07,&h07,&h5d,&hca,
       &h02,&h00
PL 120 CALL DEALLOC(DMEMORY)
LA 130 END
```

Meet ED The AmigaDOS Editor

Christopher J. Flynn

AmigaDOS—the command-driven operating system which underlies the graphics-oriented Workbench—contains two text editors. Although they aren't full-fledged word processors, these editors are ideal for entering program source code, creating batch files, and even writing short documents. This article shows how to use ED, the more powerful of the two editors. For more information on AmigaDOS and batch files, see "Introduction To AmigaDOS," a two-part series in the January and February 1986 issues of COMPUTE!, *and "AmigaDOS Batch Files," April 1986.*

The Amiga comes with more software than most people realize. Besides Amiga BASIC, Electronic Arts' *Kaleidoscope*, Mindscape's *Amiga Tutor*, the RAM disk, the speech synthesizer, the printer drivers, the icon editor, the calculator, the clock, and numerous demo programs, there are also three complete text editors. Most people know about the Notepad because it's available from the Workbench. But the other two text editors—ED and EDIT—don't show up as icons and must be run from an AmigaDOS CLI (Command Line Interface) window.

The most powerful of these text editors is ED. Although it doesn't handle multiple fonts and styles like the Notepad, it has many more editing functions and is the ideal tool for writing AmigaDOS batch files or program source code. EDIT, on the other hand, is a little more specialized. It is a sequential file editor. In practice, EDIT is best used to make changes to an existing disk file. You'll probably prefer to use ED for composing new text.

We'll be exploring ED version 1.10. Future releases of ED may change things around a little and introduce new features, so keep this in mind.

Starting ED

Where is ED hiding? Even if you peek through every nook and cranny of the Workbench, you will not find an icon for ED. It turns out that ED is actually an AmigaDOS command. This means that you have to start ED from a CLI window.

If you've never used a CLI window before, your first step will be to activate the CLI. Open the Workbench and check the contents of the System drawer. If CLIs are activated, you'll see a cube-shaped icon labeled CLI in this drawer. If the icon is not present, point to the Preferences icon and double-click the mouse's left button. Look for the CLI On/Off selector on the Preferences screen and click on the On box, then exit Preferences by specifying Save (*not* Use). Now when you reopen the System drawer, it should contain a CLI icon. If not, go back to Preferences and make sure CLI is turned on. (If you find yourself using the CLI often, you may want to drag the CLI icon from the System drawer into the main Workbench window to avoid the extra step of opening the System drawer.) To open a CLI window, double-click on the CLI icon. Now you'll have a window in which you can type AmigaDOS commands.

ED can be started in two ways:

ED *filename* [SIZE *n*]

RUN ED *filename* [SIZE *n*]

The first method starts ED from the CLI which you've just activated. It ties up the CLI until you're finished with ED. In other words, you have to leave ED before issuing other AmigaDOS commands. When you specify RUN ED, AmigaDOS automatically starts another CLI task for you and starts ED in this new CLI. Thus, you can temporarily suspend ED by moving the mouse to another window. You can go back to the original CLI and issue other AmigaDOS commands. If you are adventuresome, you can even have multiple ED sessions in progress at the same time. (What you're really doing is multitasking more than one AmigaDOS simultaneously.)

In either case, the ED command requires a filename. You can either supply the name of an existing disk file you wish to edit, or create a new file by specifying a new filename. Remember that Amiga filenames can be up to 30 characters long. So, choose filenames that take advantage of this feature. It helps you recognize your files later on.

There is a SIZE option for the ED command. (Don't type in the brackets, by the way. Brackets just signify options.) A text document must be able to fit entirely in memory. ED just cannot handle a document partly on disk and partly in memory. The SIZE option gives you a way of telling ED how much memory you want to set aside for working on the document. If you don't type in SIZE, ED will set aside 40K for you. The maximum SIZE is determined by the amount of memory you have.

Here are a few examples of commands for starting up ED:

```
ED GROCERY-LIST

ED WAR-PEACE-BOOK-REPORT SIZE
  90000
```

When SIZE is used, type out the number. Note that 90,000 bytes is typed as 90000 and not as 90,000 or 90K.

Leaving ED

When ED has been successfully started, its display occupies the entire screen. So, how can you return

to the CLI? There is no close gadget on ED's window. There is nothing to point at and click. Instead, ED requires either a Quit or an Exit command. Press the ESC (escape) key. An asterisk appears on the last line of the display. Type either Q for Quit or X for Exit and then RETURN. That's all there is to it.

There is a difference between Quit and Exit. Q leaves the editor without saving the document to disk. Anything you have typed will be lost. ED recognizes that this can be quite an inconvenience, so if you do type Q, ED displays the following warning message:

Edits will be lost - type Y to confirm:

Pressing Y at this point gets you out of ED, and no text is saved. If you type anything else, ED lets you continue working on your document.

ESC-X, the Exit option, *does* save the document on disk, using the filename you specified when you started ED. No messages are given. When ED finishes, you're back in the CLI and can then proceed with other AmigaDOS commands. When you're finished with the CLI, type ENDCLI. If you've got only one CLI window running, this returns you to the Workbench.

ED Commands

There are two types of editor commands in ED. The more direct ones are called *immediate commands* because you can enter them while typing text. Examples are line insertions and deletions. Immediate commands are always CTRL key combinations. The other category—*extended commands*—can be typed only when in the command mode. ESC-Q and ESC-X are examples. Pressing ESC opens the lowest display line on your screen for these extended commands.

When ED starts, it positions the cursor at the upper-left corner of the screen. If you are working on a new document, the screen is blank. Otherwise, the screen shows the first page of the document.

If you're creating a new document, just start typing. Notice what happens when the text approaches the right side of the screen. If a word is too long to fit on the remainder of the line, ED pulls the word down to the next line. You can keep typing without being concerned about hitting RETURN at the end of a line as you would on a typewriter.

There are several ways of correcting typos. The BACK SPACE key deletes the character to the left of the cursor. DEL deletes the character under the cursor. Table 1 lists other ways of deleting text.

ED is a full-screen editor, so you can move the cursor wherever you want with the arrow keys. To insert text, position the cursor at the desired location and begin typing. Notice that ED does not have a strikeover mode. Unwanted text has to be deleted—you can't just type over it.

The Insertion Gotcha

Try typing a few fairly long lines. Now, move the cursor to the beginning of the text. Start typing again. The existing text on the current line is moved to the right off the edge of the screen. During insertions, ED neither brings the excess text down to the next line nor enforces margins.

The disappearing text is not lost, however. ED has made one long line. The long line can be split at any point by placing the cursor where you want and pressing RETURN. If you're working with ordinary text, not source code or batch files, this may leave gaps of several spaces between sentences. To clean up the appearance, the extra spaces will have to be removed. Some other lines may need adjusting as well.

Using The Extended Commands

Extended commands (Table 2) can be typed only when ED is in the extended command mode, entered by pressing the ESC key. The cursor appears on the last line of the display. At this point, you can type one or more extended commands. It's quite handy to be able to give ED a series of commands separated by semicolons (;). When you press

Table 1: ED Immediate Commands

Command	Description
Special Keys	
BACK SPACE	Deletes the character to the left of the cursor.
DEL	Deletes the character under the cursor.
ESC	Switches to extended command mode.
RETURN	Ends the line at the cursor and starts a new line.
TAB	Moves the cursor right, adding spaces, to the next tab position.
up-arrow	Moves the cursor up one line.
down-arrow	Moves the cursor down one line.
left-arrow	Moves the cursor one character position to the left.
right-arrow	Moves the cursor one character position to the right.
Control Key Combinations	
CTRL-A	Inserts a line after the line on which the cursor is located.
CTRL-B	Deletes the line on which the cursor is located.
CTRL-D	Scrolls the text down 12 lines toward the beginning of the document.
CTRL-E	If the cursor is at the top of the screen, moves the cursor to the bottom of the screen. If the cursor is at the bottom of the screen, moves the cursor to the top of the screen.
CTRL-F	Switches the case (upper to lower or lower to upper) of the character under the cursor.
CTRL-G	Repeats the last extended command which was issued.
CTRL-H	Deletes the character to the left of the cursor. Equivalent to the BACK SPACE key.
CTRL-I	Moves the cursor right to the next tab position. Equivalent to the TAB key.
CTRL-M	Equivalent to the RETURN key.
CTRL-O	If the cursor is on a nonblank character, deletes all characters from the cursor to the first space. If the cursor is on a space, deletes all spaces from the cursor to the first nonblank character.
CTRL-R	Moves the cursor left to the first space after previous word on the current line.
CTRL-T	Moves the cursor right to the first character of the next word on the current line.
CTRL-U	Scrolls the text up 12 lines toward the end of the document
CTRL-V	Redisplays (Verifies) the screen. Insures that all the text is visible and is useful after moving or sizing the display window.
CTRL-Y	Deletes all characters on the line starting with the character under the cursor.
CTRL-[	Switches to the extended command mode. Equivalent to the ESC key.
CTRL-]	If the cursor is at the start of the line, moves the cursor to the end of the line. If the cursor is at the end of the line, moves the cursor to the start of the line.

RETURN, ED acts on the command or commands you've requested.

Extended commands can move the cursor, mark blocks of text for certain operations, and perform searches and exchanges. Some of the operations are tricky and require care. Cursor commands apply only to the cursor position in the text and not to the command line. This is fine except that you can't see the cursor in the text. You have to remember where the cursor is before you use some of the extended commands.

Sections of text can be marked by block start (BS) and block end (BE) commands. Blocks can be deleted, copied elsewhere in the document, or saved to disk. Marking a block involves moving the cursor to the first line in the block and executing the BS extended command. The end of the block is marked similarly with BE. Unfortunately, there is no visible indication of the defined text. Be very careful of cursor movements. The only help ED offers is the show (SH) command. It displays the first and last line of the block and some other information.

Text search and exchange operations work without a hitch. You can search forward (F) or backward (BF) through the document. You can exchange (E or EQ) one text string for another. Lowercase text can be treated as matching uppercase text (UC), or it can be treated as not matching (LC).

The repeat (RP) command is often used for exchanges. RP causes the command following it to be executed repeatedly until something (an error, for example) stops it. Thus, RP E carries out multiple exchange operations. Here is an example:

```
T; RP E /Compute/COMPUTE/
```

Here, the typing of COMPUTE! is being corrected. T moves the cursor to the top of the document so that the entire document will be examined. RP precedes the exchange command. Note that the two text strings are delimited by slashes. This is ED's convention when text strings are used. A "Search failed" error occurs when *Compute* can no longer be found in the text. This halts the repeat command, and the entire document will have been corrected.

The save command (SA) saves the document to disk without exiting ED. You should do this periodically to prevent disasters in the event of a power failure.

Overall, ED is an excellent general-purpose text editor. You can use it when programming, since it works with any language that accepts ASCII text files as input (including Amiga BASIC). ED can also prepare data files or help you write short letters and notes. It's not a fancy word processor, but it can handle smaller, less complex tasks quite well. ©

Table 2: ED Extended Commands

Note: /s/ refers to a single text string (/this is a string/).
/s/t/ refers to two text strings (/brown/blue/).

Command	Description
A /s/	Inserts the string on a new line after the current line.
B	Moves the cursor to the end (bottom) of the document.
BE	Places an end-of-block marker at the cursor.
BF /s/	Searches the document for the string going in a direction from the cursor toward the beginning of the document (backward find).
BS	Places a start-of-block marker at the cursor.
CE	Moves the cursor to the end of the current line.
CL	Moves the cursor one character position to the left.
CR	Moves the cursor one character position to the right.
CS	Moves the cursor to the start of the line
D	Deletes the current line. Moves all following lines up.
DB	Deletes the text marked by start-block and end-block markers.
DC	Deletes the character at the current cursor position.
E /s/t/	Replaces (exchanges) occurrences of the first string with the second string.
EQ /s/t/	The same as E, but asks you to confirm the replacement each time a match is found. Type Y or N in response to the Exchange ? prompt
EX	Extends the right margin allowing additional text to be typed.
F /s/	Searches the document for the string going in a direction from the cursor position toward the end of the document (find).
I /s/	Inserts the string on a new line before the current line.
IB	Inserts the block of text marked by start-block and end-block markers after the current line.
IF /s/	Inserts the contents of a file before the current line. The filename is given by /s/.
J	Joins the current line with the next line. This makes one new line where there were formerly two.
LC	Treats upper- and lowercase characters as different in searches.
M *n*	Moves the cursor to the line number given by *n*.
N	Moves the cursor to the starting position of the next line.
P	Moves the cursor to the starting position of the previous line.
Q	Quits ED without first saving the text. A warning message will be given stating that the text may be lost.
RP	Repeats commands. Commands are typed following RP. For example, T; RP E /brown/red/ moves the cursor to the top of the document. The Exchange command is repeated, thus changing all occurrences of *brown* to *red*. Repeat ends when an error is found. In this case, an error occurs after all the changes have been made since *brown* can no longer be found.
S	Splits the current line at the cursor location. This makes two lines where there was formerly one.
SA	Saves the document to the file specified by the original ED command. Use SA periodically to make sure you have a good copy of the text on disk.
SB	Shows the text block marked by start-block and end-block markers. The block (and any following text) will be displayed starting at the top of the screen.
SH	Shows the filename, tab distance, margin settings, first and last line of any marked text block, and the buffer full percentage.
SL *n*	Sets the left margin to the position specified by *n*. SL affects the margin setting for the entire document. New text will be typed within the margins. Existing text is not automatically reformatted when the margins change.
SR *n*	Sets the right margin to the position specified by *n*. SR affects the margin setting for the entire document. New text will be typed within the margins. Existing text is not automatically reformatted when the margins change.
ST *n*	Sets the distance the cursor moves when the TAB key is pressed.
T	Moves the cursor to the top of the document.
U	Undoes any changes made to the current line. This does not restore line deletes (D). It also does not work if you have moved the cursor from the current line.
UC	Treats upper- and lowercase characters as equivalent for searches (for example, *A* will match *a*).
WB /s/	Writes the text block marked by start-block and end-block markers to the file specified by /s/.
X	Exits ED first making sure that the document has been saved on disk.

Converting IBM ML To BASIC DATA

Mark Russinovich With Dennis Moul

This short utility converts object code created with a machine language assembler into DATA statements ready to be merged with a BASIC program. It works on any IBM PC, PCjr, or compatible with DOS 2.0 or higher.

An efficient way of speeding up crucial parts of BASIC programs or performing operations not possible in BASIC is to write a machine language subroutine. Usually, the machine language (ML) routine is loaded from disk by the BASIC program or is encoded in BASIC DATA statements that are POKEd into memory. The latter method has the advantage of making the BASIC program a stand-alone unit, not dependent on other files that must be on the same disk. Its major disadvantage, though, is that if the ML routine is more than a few bytes long, the job of converting the object code to DATA is extremely tedious and error-prone. One minor mistake could mess up the whole routine and possibly crash the system.

The solution is Program 1 below, "BIN2DAT." It takes an ML (binary) file on disk and converts it to DATA statements, ready to be merged into your BASIC program. It is impeccably reliable and takes only seconds to do its work.

Using BIN2DAT

After typing in Program 1 and saving it on disk, make sure that the ML object file you wish to convert into DATA statements is stored on disk in the .COM format. This is necessary because .EXE files have relocation information used by DOS when they are loaded into memory. Since DOS isn't used when a BASIC program POKEs an ML routine into memory, an .EXE program would not be relocated and therefore would not execute. If you've already written an .EXE file that you wish to convert to DATA statements, convert it to .COM format by using the EXE2BIN program included on the PC-DOS disk.

Now follow these steps:

1. Run BIN2DAT. It asks you for the filename of the .COM file you wish to convert. Enter the filename and press Enter. As an extra safeguard, BIN2DAT makes sure that the file has a .COM extension.

2. BIN2DAT prompts you for the output filename (the file that will contain the DATA statements). If you simply press Enter here, the filename defaults to the one displayed within brackets.

3. Next, you're asked for the starting line number of the DATA statements. Again, a default, which is line 100, is printed within brackets. Either press Enter or type your own starting line number.

4. BIN2DAT now asks for the line number increment (the default is 10) and the numeric base of the data—decimal or hexadecimal. The base makes little difference, but the default is hexadecimal because sometimes it's useful to compare the .LST file generated by the assembler with the DATA statements.

Merging The DATA

Once you've entered all the required information, BIN2DAT creates the BASIC data file to your specifications. To merge it with your BASIC program, load the BASIC program and type:

```
MERGE "filename.ext"
```

You'll notice that the first line in the file has only one data value. This isn't part of the ML. This value is the size of the ML routine in bytes, minus one. Therefore, it corresponds to the upper limit of a FOR-NEXT loop that is required to POKE the ML routine into memory.

Next are the lines containing the data for the ML program. An example of an ML routine is seen in Program 2, "EXAMPLE.ML." Program 3, "Demo DATA," shows the file produced by BIN2DAT after converting the .COM file produced by an assembler and EXAMPLE.ML. Extra lines have been added to POKE the ML routine into memory and CALL it. Examining these listings should clear up any questions about how to use BIN2DAT.

How It Works

BIN2DAT is fairly straightforward. Once all the information has been entered by the user, the SHELL command is used to create a file with the directory entry of the ML file. SHELL allows the use of DOS commands from BASIC, but in the DOS 2.0/2.1 generation, it has the flaw of altering memory locations 30H and 31H, which happen to point to the beginning of the BASIC program in memory. To overcome this, the values for these locations are PEEKed before the SHELL command is executed and then POKEd back later.

The next part of the program reads the size of the ML file out of the directory random file which was made by SHELL. Then it begins constructing the DATA statements, which are sent to the output file. The first DATA line has only the count value (described above). Subsequent lines have ten data numbers each. The MOD 10 function checks for the end of a line. When a line ends, it is sent to the output file and a new line is started.

After the ML program has been completely read and the new file is finished, the CLOSE command closes the input and output files, and the program terminates.

Several changes can make BIN2DAT serve your particular needs better. If you usually start your data on some line other than 100, this default value can be changed. Also, the default values for the line increment and numeric base can be changed to make running the program easier. If you want to have more than or fewer than ten items per data line, you can change the number 10 in each MOD function to some other number.

Program 1: BIN2DAT

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!

```
LJ 10 DEF SEG
BH 20 KEY OFF
GO 30 ON ERROR GOTO 570
OM 40 :
CF 50 REM Print title and get in
      fo
OO 60 :
KN 70 PRINT "Binary to Data Stat
      ement Converter"
NM 80 PRINT "(c) Copyright 1986,
       Compute' Publications"
FL 90 PRINT
OD 100 INPUT "File to convert: "
      ,FSOURCE$
BF 110 IF INSTR(FSOURCE$,".COM")
      =0 AND INSTR(FSOURCE$,".c
      om")=0 THEN  PRINT:BEEP:P
      RINT "File must have .COM
       extension.":END
DQ 120 FILEN=INSTR(FSOURCE$,".")
      -1:FILEN$=LEFT$(FSOURCE$,
      FILEN)+".BAS"
PL 130 PRINT "Data file [";FILEN
      $;"]";:INPUT ": ",FDEST$
NF 140 IF FDEST$="" THEN FDEST$=
      FILEN$
HF 150 INPUT "Starting line numb
      er [100]: ",SLN
ED 160 INPUT "Line increment [10
      ]: ",LINC
JF 170 INPUT "Hex/decimal [h]: "
      ,H$
FD 180 IF SLN=0 THEN SLN=100
HC 190 IF LINC=0 THEN LINC=10
GM 200 IF H$="" OR H$="h" OR H$=
      "H" THEN H=1
MG 210 :
NG 220 REM Capture directory in
      random file
NK 230 :
JA 240 P1=PEEK(&H30):P2=PEEK(&H3
      1)
NC 250 SHELL "dir "+FSOURCE$+" >
      $$zztemp"
MG 260 POKE &H30,P1:POKE &H31,P2
DB 270 OPEN "$$zztemp" FOR INPUT
       AS 2
DJ 280 FOR I=1 TO 4:INPUT#2,DMY$
      :NEXT
JH 290 INPUT#2,ENTRY$
PL 300 REM Get size of com file
      from dir
KO 310 SIZE=VAL(MID$(ENTRY$,16,6
      ))
ND 320 CLOSE #2 :KILL "$$zztemp"
NL 330 :
EL 340 REM Open com file and new
       dat file
NP 350 :
NF 360 OPEN FSOURCE$ AS 1 LEN=1
MD 370 FIELD 1,1 AS BYTE$
IF 380 OPEN FDEST$ FOR OUTPUT AS
       2
LO 390 LINNUM=SLN+LINC
HF 400 LIN$=STR$(SLN)+" DATA"
```

Program 2: EXAMPLE.ML

Note: This source code listing is for illustrative purposes only. It requires an assembler to enter.

```
; This is a sample assembly
; language program that will
; be poked in and run from
; BASIC.

prog    segment

        assume  cs:prog,ds:prog

main    proc    far
        push    ax
        push    bx
        push    dx

; Print characters

        mov     bx,offset mess1
print:  mov     dl,cs:[bx] ;get char
        cmp     dl,0    ;are we through?
        je      exit    ;yes, return
        inc     bx      ;no, get nxt char
        mov     ah,2    ;dos print routine
        int     21h
        jmp     print   ;get more
exit:   pop     dx      ;restore stack
        pop     bx
        pop     ax
        ret

mess1   db 0dh,0ah,'This is output of a'
        db ' sample assembly language'
        db ' program.',0dh,0ah,0dh,0ah
        db 0dh,0

main    endp

prog    ends
        end     main
```

```
IE 410 IF H=1 THEN L1N$=L1N$+" &
      h"+HEX$(SIZE-1) ELSE L1N$
      =LIN$+STR$(SIZE-1)
MH 420 PRINT#2,LIN$
PJ 430 LIN$=STR$(LINNUM)+" DATA
      "
CM 440 LINNUM=L1NNUM+L1NC
EH 450 FOR COUNT=1 TO SIZE
BH 460   GET #1,COUNT
DM 470   WBYTE$=BYTE$
CC 480   IF H=1 THEN NUM$="&h"+H
      EX$(ASC(WBYTE$)) ELSE NUM
      $=STR$(ASC(WBYTE$)):
            NUM$=RIGHT$(NUM$,L
      EN(NUM$)-1)
LI 490   IF COUNT MOD 10>0 THEN
      520
DB 500   PRINT#2,L1N$+NUM$:L1N$=
      STR$(LINNUM)+" DATA "
PC 510   L1NNUM=L1NNUM+LINC:GOTO
       530
MF 520   L1N$=LIN$+NUM$+","
MH 530 NEXT
MP 540 IF COUNT MOD 10<>1 THEN L
      1N$=LEFT$(L1N$,LEN(L1N$)-
      1):PRINT#2,LIN$
PE 550 CLOSE
FD 560 PRINT:PRINT "File written
      ":END
PD 570 BEEP:PRINT "DOS error - a
      borting.":CLOSE:END
```

Program 3: Demo DATA

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing in Programs" in this issue of COMPUTE!

```
GC 10 REM  This program will pok
      e in an
HF 20 REM  assembly language pro
      gram and
LL 30 REM  then CALL it.
DM 40 :
MD 50 DEF SEG=&H1700
QN 60 READ COUNT
EB 70 FOR MEM=0 TO COUNT
JP 80   READ BYTE
LB 90   POKE MEM,BYTE
MN 100 NEXT
MF 110 :
KC 120 SAMPLE=0
QI 130 CALL SAMPLE
LC 140 END
MM 150 :
CM 160 DATA &h55
QP 170 DATA &h50,&h53,&h52,&hBB,
      &h19,&h0,&h2E,&hBA,&h17,&
      hB0
LM 180 DATA &hFA,&h0,&h74,&h7,&h
      43,&hB4,&h2,&hCD,&h21,&hE
      B
HG 190 DATA &hF1,&h5A,&h5B,&h5B,
      &hCB,&hD,&hA,&h54,&h6B,&h
      69
KE 200 DATA &h73,&h20,&h69,&h73,
      &h20,&h6F,&h75,&h74,&h70,
      &h75
FD 210 DATA &h74,&h20,&h6F,&h66,
      &h20,&h61,&h20,&h73,&h61,
      &h6D
JM 220 DATA &h70,&h6C,&h65,&h20,
      &h61,&h73,&h73,&h65,&h6D,
      &h62
PG 230 DATA &h6C,&h79,&h20,&h6C,
      &h61,&h6E,&h67,&h75,&h61,
      &h67
DE 240 DATA &h65,&h20,&h70,&h72,
      &h6F,&h67,&h72,&h61,&h6D,
      &h2E
CD 250 DATA &hD,&hA,&hD,&hA,&hD,
      &h0
```


64 Fleet List

Buck Childress

Have you ever wished you could zip forward or backward through a program listing at the touch of a key? That capability is especially valuable when you're writing or debugging a long BASIC program. This Commodore 64 utility lets you do exactly that—scroll a program listing up or down on the screen using the 64's special function keys.

"Fleet List" simplifies and speeds up the process of editing a BASIC program listing. As a bonus, it can also tell you the current number of lines in a program and is very easy to use.

Since Fleet List is written entirely in machine language, it must be entered using the "MLX" machine language entry program, published elsewhere in this issue. Be sure you have read and understood the instructions for using MLX before you begin entering the data for Fleet List. When you first run MLX, you'll be asked for starting and ending addresses. The proper values for Fleet List are as follows:

Starting address: C000
Ending address: C367

After you have entered all the data for Fleet List, be sure to use the MLX Save option to store at least one copy of the data before proceeding.

Scroll In Either Direction

To use Fleet List, load it into memory with LOAD *"filename"*,8,1 (for tape, change the ,8,1 to ,1,1), then type NEW and press RETURN to reset memory pointers. Fleet List is now in memory, but it's not active yet. You should first load the BASIC program you wish to edit, then type SYS 49152 and press RETURN to activate Fleet List. (For the utility to function properly, there must be a BASIC program in memory when Fleet List is activated.) It can handle programs up to 1,600 lines in length (a warning is issued if your program is too long).

To scroll the listing forward, press the f1 function key (the text will be dark gray). To scroll backward, press f3 (the text will be black). To move quickly from one part of the program to another, hold down the Commodore key while pressing f1 or f3. You'll see the line numbers spin past on your screen. When you release the Commodore logo key, Fleet List begins listing from that point onward. At other times you may want a slow-motion listing. To slow down the

scrolling in either direction, press f2 (SHIFT-f1) or f4 (SHIFT-f3).

If you scroll past the end or beginning of your program, Fleet List simply wraps around to the other end of the program. For instance, say that your program starts with line 10 and ends at line 1000. If you scroll forward past line 1000, Fleet List prints a line on the screen as a marker and then begins to list forward from line 10. If you scroll backward past line 10, Fleet List prints a marker line and begins to work downward from line 1000.

Fleet List also provides an easy way to move immediately to the beginning or end of the program. If you press the f7 key, the list starts at the first line in the program. Remember, Fleet List wraps around the ends of the program automatically, so to get to the very last line, simply scroll backward one line from the beginning.

Line Count

You can find out how many lines you have in your program at any time by pressing the f5 function key. When Fleet List is first activated, it also displays the number of lines in whatever program is currently in memory. As you add and delete lines, the f5 key comes in handy.

Of course, if you're writing a program that uses the function keys for its own purposes, you want to be able to enter them normally in a program line. To allow for this possibility, Fleet List checks for quote and insert modes and does not respond when you're in either mode. When you leave quote or insert mode (usually by pressing RETURN), Fleet List is active again.

Fleet List does not interfere with the process of editing existing program lines, entering new ones, or moving around on the screen with the cursor keys. And there's no need to clear the screen to relist after such activities. Before it begins to list again, Fleet List automatically positions the cursor at the bottom of the screen.

Because Fleet List resides in a memory area that's not normally used by BASIC, you should be able to load and save BASIC programs without disturbing it. However, before loading or saving, it's a good idea to deactivate Fleet List by pressing RUN/STOP-RESTORE. To reactivate Fleet List, type SYS 49152 and press RETURN.

Fleet List

For instructions on entering this listing, please refer to the "MLX" article in this issue of COMPUTE!

```
C000:AD 14 03 AE 15 03 C9 5E 4F
C008:D0 04 E0 C0 F0 4F 8D 70 6B
C010:C3 0E 71 C3 20 B0 C2 0C S7
C010:73 C3 0C 74 C3 8C 76 C3 1E
C020:0C 78 C3 8C 79 C3 8C 7E B9
C020:C3 A9 0F A2 0B 0D 20 D0 A1
C030:0D 21 D0 8E 86 02 8E 7S 92
C038:C3 20 44 E5 A9 0E 20 D2 25
C040:FF A9 00 20 D2 FF A0 00 07
C040:20 FB C0 20 18 C3 A9 5E 74
C0S0:0D 7E C3 A2 C0 78 8D 14 F1
C050:03 8E 15 03 S8 60 AD 74 ES
C060:C3 D0 6D AS D4 D0 66 AS SC
C068:D8 D0 62 AD 8D 02 C9 03 BC
C070:B0 5E AS C0 C9 40 F0 S8 DC
C070:C9 01 D0 03 0D 79 C3 C9 0D
C080:03 D0 0C 8D 74 C3 20 B0 BS
C088:C2 0C 76 C3 4C A3 C2 C9 D9
C090:06 D0 32 8D 74 C3 8D 75 AB
C098:C3 A0 00 8C 7E C3 AS FB 37
C0A0:48 A5 FC 48 A9 01 20 FB 61
C0A0:C0 60 05 FC 60 0S FB A2 19
C0B0:10 06 CC 8E 7E C3 A0 00 A6
C0B8:10 20 F0 FF 20 4D C3 20 4A
C0C0:18 C3 4C A3 C2 C9 04 90 D0
C0C0:04 C9 06 90 06 0D 78 C3 A3
C0D0:6C 70 C3 0D 74 C3 AD 70 7C
C0D0:C3 F0 13 A9 00 8D 73 C3 56
C0E0:8D 78 C3 A2 10 A0 00 18 4S
C0E0:20 F0 FF 20 4D C3 AC 73 FE
C0F0:C3 D0 0S AD 79 C3 D0 03 03
C0F8:4C 70 C1 8C 7C C3 0C 7D 0A
C100:C3 A5 FD 48 A5 FE 40 20 EC
C100:B0 C2 A0 01 84 CC 00 01 D2
C110:FB F0 06 20 C4 C2 4C 0A 64
C110:C1 20 C4 C2 20 C4 C2 01 94
C120:FB F0 1E 20 C4 C2 A5 FB 1C
C128:A6 FC 91 FD 20 C0 C2 8A 90
C130:91 FD 20 CB C2 20 E9 C2 E9
C130:20 C4 C2 20 C4 C2 4C 0A 2B
C140:C1 20 F2 C2 A5 FD A6 FE A0
C148:8D 7A C3 8E 7B C3 A0 00 BE
C150:A5 FE C9 D0 00 16 E6 FD 56
C158:D0 08 E6 FE A5 FE C9 D0 A0
C160:B0 0A AS FE 85 CC 90 91 0S
C160:FD 4C S6 C1 20 B0 C2 60 96
C170:05 FE 68 8S FD AD 7E C3 43
C178:D0 01 60 A0 01 04 CC 80 F6
C100:AD 74 C3 C9 04 F0 20 8C D3
C100:06 02 AD 76 C3 C9 02 D0 07
C190:06 20 F2 C2 20 F2 C2 A9 A5
C198:01 0D 76 C3 20 F2 C2 20 7D
C1A0:F2 C2 A0 01 B1 FD F0 F4 CE
C1A0:0S FC 00 01 FD 85 FB 4C A4
C1B0:E2 C1 A9 0B 0D 06 02 AD 34
C1B0:76 C3 C9 01 D0 06 20 CB SC
C1C0:C2 20 CB C2 A9 02 8D 76 3A
C1C8:C3 B1 FD 05 FB 20 C0 C2 6D
C1D0:01 FD D0 09 20 B0 C2 20 C0
C1D8:13 C3 4C C9 C1 8S FC 20 30
C1E0:CB C2 AD 0D 02 40 40 8D D0
C1E8:77 C3 C9 02 D0 08 20 4D A6
C1F0:C3 A9 91 20 D2 FF B1 FB EA
C1F0:AA 20 C4 C2 B1 FB 20 CD 2A
C200:BD 20 C4 C2 A9 20 20 D2 12
C200:FF 60 AA E0 02 D0 08 A9 10
C210:20 20 D2 FF 20 D2 FF A0 F4
C210:00 01 FB F0 3D E0 02 D0 DA
C220:06 20 C4 C2 4C 17 C2 C9 03
C220:00 90 26 A6 D4 D0 22 30 A7
C230:E9 7F AA A0 FF CA F0 00 FE
C238:C0 B9 9E A0 10 FA 30 F5 31
C240:C8 09 9E A0 C9 80 00 06 2E
C248:20 D2 FF 4C 40 C2 38 E9 0E
C2S0:80 20 D2 FF 20 C4 C2 4C 5E
C2S8:17 C2 68 C9 02 F0 03 20 0D
C260:4D C3 A5 C0 C9 04 90 31 9F
C260:C9 06 B0 2D 8D 74 C3 AD 00
C270:8D 02 C9 02 F0 0A AD 77 19
C278:C3 C9 02 D0 06 20 4D C3 AE
C280:4C 7B C1 AD 8D 02 F0 F8 6D
C280:C9 02 B0 F7 A2 00 A0 00 5F
C290:E8 D0 FD C0 D0 FA 4C 7B 91
C298:C1 AD 77 C3 C9 02 D0 03 90
C2A0:20 4D C3 A9 00 85 C6 8D CD
C2A8:74 C3 0D 79 C3 6C 70 C3 17
C200:A0 00 A2 00 84 FB 86 FD 02
C2B8:A9 08 A2 C3 8D 73 C3 85 EC
C2C0:FC 86 FE 60 E6 FB D0 02 17
C2C0:E6 FC 60 E6 FD D0 F0 E6 8D
C2D0:FE A5 FE C9 D0 90 F3 20 8C
C2D0:44 E5 20 13 C3 A2 08 20 08
C2E0:41 C3 20 30 C3 C0 00 F0 11
C2E8:FC EE 7C C3 D0 DC EE 7D C9
C2F0:C3 60 A5 FE C9 C4 B0 06 DD
C2F8:AS FD C9 81 90 00 C6 FD 5E
C300:AS FD C9 FF D0 02 C6 FE 2E
C308:60 AD 7A C3 AE 70 C3 85 27
C310:FD 86 FE A9 00 8D 75 C3 97
C318:20 4D C3 A2 28 A9 2A 20 02
C320:D2 FF CA D0 FA AD 75 C3 B4
C328:F0 23 8E 75 C3 20 4D C3 17
C330:AE 7C C3 AD 7D C3 20 CD 0A
C338:0D A2 00 20 41 C3 4C 18 13
C340:C3 BD 5S C3 F0 0E 20 D2 D2
C340:FF E0 4C 41 C3 A9 0D 20 A6
C3S0:D2 FF A0 00 60 20 4C 49 BA
C350:4E 45 53 0D 00 4F 56 4S C2
C360:52 20 00 00 00 00 00 00 19
```

©

Automatic Typist: Using Apple Exec Files

Mike Miyake

Although it's often overlooked, the EXEC command offers an easy way to extend the power of Applesoft BASIC. EXEC can read and perform commands directly from a disk file, just as if you'd typed the commands on the keyboard yourself. It can also be used as a convenient, built-in merge command for adding frequently used subroutines to Applesoft programs. The example programs below run on any Apple II-series computer; most work with either DOS 3.3 or ProDOS. A disk drive is required.

Have you ever wanted to know the address of a machine language program, or the number code for one of the Apple's 16 low-resolution colors? Are you curious about how a particular Applesoft program uses the computer's memory? In most cases finding the answers to such questions means thumbing through a reference book or typing cumbersome statements like PRINT PEEK (N)+256*PEEK(N+1) to examine memory. And the PEEK statement usually must be typed in immediate mode, since running a short program to get at the information would destroy any program that's already in memory.

"Automatic Typist" shows you how *exec files* can solve such problems. An exec file is simply a text file which you activate with an EXEC command from Applesoft BASIC. It executes like an immediate mode statement—something you type directly on the keyboard, without a line number—but it can also be saved to disk and reused over and over, just like a program. In effect, exec files let you control the computer with disk files that act like immediate commands without disturbing a program that's in memory. We'll show how to put both immediate mode commands and program lines in exec files, and provide some useful examples of what exec files can do.

Creating Exec Files

Type in Program 1, then run it once to make sure it works correctly. Run the program and follow the prompts, entering any filename when prompted. Since this is just for practice, it doesn't matter what filename you use. After that's done, exit the program and type CATALOG to view the new file; it should show up as a text (T) file. Now delete the file (it doesn't contain any data, so you're not losing anything of importance).

Once you're satisfied that Program 1 works correctly, it can be used to create exec files. An exec file ordinarily contains one or more statements in the form of ordinary text. Unfortunately, the Apple II *DOS Manual* tells you very little about how to create such a file. In most cases the simplest way to do so is within a BASIC program. Program 1 illustrates the basic technique. Once the file has been opened (line 18) and a WRITE statement has executed (line 20), all subsequent PRINT and LIST statements send their output to the disk file instead of to the screen. Other BASIC statements function normally while the output of PRINT and LIST is being diverted. When the file is closed (line 1000), PRINT and LIST resume their normal functions.

Program 1 provides you with a template program for creating exec files. It lets you choose a filename, opens the file, and prepares it for writing. To use this program, you need only add appropriate PRINT and/or LIST statements in new lines between lines 20 and 1000 of the template.

Let's try a simple example. Load Program 1, then type in the lines listed in Program 2. The object is to add the lines from Program 2 to the template program. The initial PRINT statements in lines 50–70 write the commands bracketed inside quotes to the disk file. In cases where the exec file itself will contain PRINT statements, it's necessary to write quote characters to the file as well. This is done in line 70 with the variable Q$, which Program 1 defines as CHR$(34) in line 10.

After you finish adding the lines from Program 2, run the program. Enter the filename CC when asked for a filename, then press the space bar when prompted. The text inside quotes in lines 50–70 is written to a disk file named CC. To execute this file, exit the program and type EXEC CC. It displays all 16 lo-res colors in vertical bars on the screen, with a matching number code directly beneath each color bar.

If the CC file doesn't work properly, delete it and repeat the process. If you want to use the same filename (the normal case), it's necessary to delete the old version of a text file before writing an updated version of the file to disk. Unlike

BASIC program files, which automatically replace an existing file with a new file of the same name, text files simply append new information to the end of the existing file.

A Program-Writing Program

The previous example printed immediate mode statements (commands without line numbers) to the exec file. But you can also print numbered program lines to a file. For instance, reload Program 1 and add this line:

```
40 PRINT"100 TEXT:HOME"
```

Run the program and write the file to disk using the filename HOMER. It creates a text file consisting of the BASIC program line 100 TEXT:HOME (we'll explain below why you might want to create this type of file). Although you can write program lines to a disk file with PRINT, it's often more convenient to use LIST instead. One advantage of doing so is that you can type the lines exactly as they normally appear without having to enclose everything in PRINT statements.

To illustrate, let's create the HOMER exec file with LIST instead of PRINT. DELETE the HOMER file from your disk, then reload Program 1 and enter these new lines:

```
21 REM CAPTURE BASIC
22 REM PROGRAM LINES
23 LIST 24,999: GOTO 1000
100 TEXT:HOME
```

The LIST command in line 23 writes every program line from 24 to 999 to the disk file. The GOTO command branches around the data (one line in this case) that we're writing to disk. One disadvantage of this technique is that the lines to be written to disk must fall between 24 and 999, inclusive. By renumbering either the template program or the lines to be written, you should be able to overcome this problem in most cases.

Merging Common Subroutines

Since the EXEC command is analogous to typing, an exec file is a good place to save commonly used subroutines for reuse in different programs. This makes it easy to merge the subroutine contained in the exec file with a program already in memory. To bring the lines into memory, simply type EXEC *filename*. As long as the exec file contains no lines numbered the same as those in the current program, the new lines are added without disturbing the program in memory.

To illustrate, let's save Program 1 in exec file form. Reload Program 1, then add this line:

```
25 LIST 1,20:LIST 1000,
```

Now run the program, entering C.LINES when prompted for a filename. Exit the program, then type NEW, followed by LIST to confirm that no program is in memory. Type EXEC C.LINES and press RETURN. Several bracket prompts will scroll past as the computer enters each program line automatically. When the cursor reappears, type LIST. Program 1 is back in memory, just as if you had typed each line manually.

It's not difficult to see how much programming time this method could save, particularly if you build up a library of commonly used subroutines that each use different ranges of line numbers.

Last BLOAD

Program 3 is an exec file that comes in handy in many different situations. Its purpose is to tell you the load address and length of the last file that was BLOADed into memory. Knowing this information lets you run machine language programs immediately with a CALL statement, or copy them without using DOS 3.3's FID utility. Unlike the other examples presented here, this one is for DOS 3.3 only—it does not work with ProDOS.

The procedure for creating this file should be familiar by now: Reload Program 1, add the lines listed in Program 3, then run the program. This exec file uses pointer locations applicable to a 48K Apple II. If you have a 16K or 32K system, change the pointer locations as shown here:

```
32K: address=PEEK(27250)+256* PEEK(27251)
     length=PEEK(27232)+256* PEEK(27233)

16K: address=PEEK(10866)+256* PEEK(10867)
     length=PEEK(10848)+256* PEEK(10849)
```

Memory Map

Program 4 contains the lines to add to Program 1 to create another useful exec file. This one shows the memory locations of the current BASIC program and its strings and variables. To use it, load and run the BASIC program you're curious about, then type EXEC *filename* in immediate mode. The pointer locations used by this file are discussed on page 140 of the *Applesoft II BASIC Reference Manual*.

For instructions on entering these listings, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!

Program 1: Exec File Maker

```
00 10 D$ = CHR$ (4):O$ = CHR$ (3
      4)
55 11 TEXT : HOME : HTAB 15: PRI
      NT "MAKE FILES:"
60 12 VTAB 6: INPUT "FILENAME: "
      ;N$: IF LEN (N$) = 0 THEN
      12
D2 14 HTAB 1: VTAB 8: CALL - 95B
FE 16 PRINT "INSERT NON-WRITE PR
      OTECTED DISK": PRINT "AND
      PRESS SPACE BAR WHEN READY
      .": PRINT : PRINT "  => ";
      : GET A$
9C 1B PRINT D$"OPEN"N$: PRINT D$
      "CLOSE"
E5 20 PRINT D$"OPEN"N$: PRINT D$
      "WRITE"N$
61 1000 PRINT D$"CLOSE"
95 1002 VTAB 14: PRINT "DO IT AG
      AIN?";: GET A$
80 1004 IF A$ = "Y" THEN 14
F2 1006 END
```

Program 2: Color Chart

```
49 REM  CC.LINES
50 PRINT "TEXT:GR"
60 PRINT "FOR I=1TO15:COLOR=1:VL
   IN20,39 AT2*I:VLIN20,39 AT 2
   *I+1:NEXT"
70 PRINT "PRINT"O$"0   2   4
   6   8   10  12  14"O$":PRINT
   "O$"  1   3   5   7   9   11
     13  15"O$
```

Program 3: Last BLOAD

```
49 REM  LAST BLOAD / 3.3 DOS/4B
   K
50 PRINT "TEXT:HOME"
100 PRINT "PRINT"O$"LAST BLOAD"
    O$":PRINT:PRINT"O$"M/L FILE
    : ADDRESS = "O$"PEEK(-21902
    )+256*PEEK(-21901):HTAB12:P
    RINT"O$"LENGTH = "O$"PEEK(-
    21920)+256*PEEK(-21919)"
```

Program 4: Memory Map

```
49 REM  MEMORY MAP
50 PRINT "TEXT:HOME"
60 PRINT "PRINT"O$"MEMORY MAP"O
   $":PRINT:PRINT"O$"HIMEM"O$":
   PRINT"O$"STRINGS (DOWN TO):"
   O$":PRINT"O$"(FREE SPACE)"O$
   ":PRINT"O$"ARRAYS, POINTERS"
   O$": PRINT "O$"& VARIABLES (
   UP TO):"O$":PRINT"O$"LOMEM"O
   $":PRINT"O$"PROGRAM LINES TO
   :"O$
70 PRINT "POKE32,22:VTAB3:PRINT
   :PRINT:PRINTPEEK(115)+256*PE
   EK(116):PRINTPEEK(111)+256*P
   EEK(112):PRINT:PRINT:PRINTPE
   EK(109)+256*PEEK(110):PRINTP
   EEK(105)+256*PEEK(106):PRINT
   PEEK(175)+256*PEEK(176)"
80 PRINT "POKE32,0"
```

Atari Password

Glenn Anderson

Would you like to protect a diskful of important programs from prying eyes? If so, here's a solution that discourages all but the most determined snoops: a security program that denies access to your disk unless the correct password is entered. Even if someone boots from a different disk and bypasses the security program, your BASIC listings remain unreadable. For all Atari 400/800, XL, and XE computers with at least 24K RAM, a disk drive, and Atari DOS 2.0, 2.5, or 3.0.

Most people at one time or another have felt the need to protect their programs from prying eyes. At first the solution seems simple: When the program starts, it can ask the user to type in a code which is then compared to a password embedded in the program. If the user fails to enter the right password, the program can end with a NEW command, erasing itself from memory.

This might deflect a rank beginner, but not many other computer users would be fooled. Anyone could obtain the password merely by stopping the program with the BREAK key, typing LIST, and reading through the listing. A password serves no purpose if it can be found so easily.

To keep people from stopping the program and scanning the listing, you can disable the BREAK key by adding this line:

```
1 POKE 16,64:POKE 53774,64
```

Now if the user hits BREAK, nothing happens.

The next thing a persistent person will do, however, is press the Atari's SYSTEM RESET button. The computer does what is called a *warm start*, and the program stops. Since the program is still in memory, the user can type LIST and start looking for the password.

To prevent this from happening, you can add this line:

```
2 POKE 580,1
```

Now when SYSTEM RESET is pressed, the computer does a *cold start* instead of a warm start. It has the same effect as switching the power off and then on again, erasing any program in memory, rebooting the disk operating system (DOS), and loading and running an AUTORUN.SYS file if one is present on the DOS disk. With BREAK and SYSTEM RESET now safely disabled or trapped, the user can't stop your program and discover the secret password.

It's Still Vulnerable

But that assumes your program is running. The user can simply load the program without running it, then type LIST. To prevent this, you could make the BASIC program run automatically on powerup by writing a machine language booter or creating an AUTORUN.SYS file with the autobooting utility included with DOS 2.5. Whenever the computer is booted with this disk, the program automatically runs, and the user must enter the correct password to gain access to the rest of the program.

This works if the user boots with that disk. But what's to stop people from booting with another disk? They can easily gain control of BASIC, insert your disk, load your program, and find the password.

What's really needed is a way to save the program so that it can be run but not loaded. A method for this has already been found and published by COMPUTE! Books in *Mapping the Atari*, and similar solutions have appeared in other publications. To protect a program from being loaded, these two lines must be added:

```
32766 FOR VARI=PEEK(130)+PEEK(131)
      *256 TO PEEK(132)+PEEK
      (133)*256:POKE VARI,155:NEXT
      VARI
32767 POKE PEEK(138)+PEEK
      (139)*256+2,0:SAVE
      "D:filename.ext":NEW
```

It is important to make these the last two lines in the program. The first line fills the variable name table with RETURN characters. The second line finds the location in memory of the current statement line—line 32767 in this example—and POKEs the value of zero into the length of that line. Now, when

the computer tries to access a statement with a line number higher than 32767, it gets caught at line 32767 when searching for the line.

This keeps the program from being loaded because of the way Atari BASIC handles an immediate mode command—it treats the immediate mode line as if it were numbered 32768. Since 32768 is higher than 32767, the computer never finds the immediate mode line and never executes it. Therefore, unless the computer is executing the program, the system is effectively crashed because nothing can be done in immediate mode. With this done, the only way to get the program into memory without crashing the system is to run the program at the same time it is loaded from disk: RUN "D:*filename.ext*".

To lock and save a program in this manner, you enter GOTO 32766. The routine saves the program with the filename you specified in line 32767. It also erases the program from memory with NEW, so it's a good idea to save an unprotected copy on another disk before protecting it in case the program needs revisions or debugging.

The Keeper Of The Keys

Now we've got the basis of a password program that can be used to keep out unwanted users. And, thanks to the AUTORUN.SYS loader, the program runs automatically when the disk is booted.

Another idea is to make this autoboot program a menu program that can run other protected programs. This saves the trouble of adding a password procedure to all the protected programs on the disk. To let the other programs know that the user has successfully entered the correct password, the menu program can POKE some arbitrary but predetermined number into any location in an area of memory that is not erased when a new program runs. Then the first line of the new program can check this location for the proper value. If the location does not hold the correct value, the program can stop with a NEW command or rerun the menu program.

You might also want to make the other programs rerun the menu program when they're finished. If this is done, it's wise to have the menu program check the secret memory location for the desired value again so that it knows whether the password has been successfully entered already. If it has, the menu program can skip over the password procedure.

"Atari Password," listed below as Program 1, does all this and a little more. It also includes a way to change the password and unprotect the program.

When typing Program 1, be especially careful with the DATA statements in lines 1–6. They contain information for restoring the variable name table when unprotecting the program.

The initial password is in line 120: ENTER. Type this line exactly as it appears. If you want to change the password later, do it with the option provided for this purpose when running Atari Password, *not* by changing line 120.

Creating A Password Disk

When you've finished typing in Program 1, follow these steps *before running the program*:

1. LIST at least one copy of Atari Password on a backup disk with the command LIST "D:*filename.ext*". Retain this copy as your unprotected backup. Use any filename you like except AUTORUN .BAS, because that's the name used by the protected version of Atari Password.

2. Don't run the program yet. After saving your backup, type NEW to erase it from memory. Then type in Program 2, "Autoboot Maker," and save at least one copy of that program on your backup disk. Don't run this program yet, either.

3. Type NEW to erase Program 2 from memory. Reload Atari Password (Program 1) from your backup disk with the command ENTER "D:*filename.ext*". This ensures that the variable name table will be in the proper state so that the program can be unprotected properly.

4. Now you're ready to create a protected version of Atari Password. Insert a formatted disk that contains Atari DOS 2.0, 2.5, or 3.0. This will be your protected password disk.

5. Type GOTO 9500 and press RETURN. After a brief pause, Atari Password saves a protected version of itself on the disk with the filename AUTORUN.BAS. When it's done, it erases itself from memory.

6. Remove the password disk and insert the backup disk. Load Program 2.

7. Remove the backup disk and insert the password disk. Run Program 2. It creates an AUTORUN.SYS file on the password disk and informs you when it's done. If you've made a typing mistake in the DATA statements, it notifies you of your error. On power-up, this AUTORUN.SYS file runs a BASIC program named AUTORUN.BAS—the protected version of Atari Password. (Note that if there's already an AUTORUN.SYS file on the disk, it will be replaced by this AUTORUN.SYS. Rename or move the existing AUTORUN.SYS to another disk if you don't want to lose it.)

8. The password disk is now prepared. To confirm that Atari Password is working properly, turn the power off, then on again to boot the disk. Atari Password should automatically load and run. You should be able to gain access to the program by typing the default password, ENTER, and then pressing the START button (do not press RETURN). Type the password carefully; the actual keys you press are not echoed on the screen, so it's easy to make a typing mistake. The SELECT button backspaces, and OPTION erases the entire input line. If you accidentally hit the CAPS key, the program may not recognize your password; uppercase and lowercase are significant. If the program denies access with a LOCKOUT message, press START to try again or SYSTEM RESET to reboot.

Using Atari Password

Once you've gained access, Atari Password presents a short menu. Press 1, 2, or 3 for your choice:

```
1 EXIT TO BASIC
2 CHANGE PASSWORD CODE
3 DISK DIRECTORY
```

Option 1 exits Atari Password, erases the program from memory,

and leaves you in BASIC.

Option 2 lets you change the password from the default—ENTER—to anything you wish. When using this option, make sure you have the password disk inserted in the drive. It rewrites the part of Atari Password which checks for the code word. You can enter any combination of letters or numbers for the password, but it should be no more than 28 characters long.

Option 3 calls up a disk directory on the screen. From this directory, you can load and run any BASIC program saved on the disk. To pick a program, move the arrow pointer with the cursor keys (you don't have to hold down CTRL as you normally do when moving the cursor in BASIC). Then press RETURN. If Atari Password can't load and run the program for some reason—perhaps it's not a BASIC program, or it's saved in LIST format—you're informed of this and allowed to pick another program. To return to the main menu, press the SELECT button.

Protecting BASIC Programs

To protect an ordinary BASIC program and make it dependent on Atari Password, follow these steps:

1. Type in these three lines and LIST them to disk:

```
0 IF PEEK(1612)<>126 THEN RUN"D:
  AUTORUN.BAS"
32766 FOR VARI=PEEK(130)+PEEK
      (131)*256 TO PEEK(132) +PEEK
      (133)*256:POKE VARI,155:NEXT
      VARI
32767 POKE PEEK(138)+PEEK(139)*
      256+2,0:SAVE"D:filename.ext"
      :NEW
```

(Notice line 0; it checks to see whether memory location 1612 contains the value 126. If not, it reruns Atari Password, a protection technique that we mentioned above. If you change Atari Password to put a different number in this location, or if you change the location, be sure to make the change here also.)

2. Load the BASIC program you want to protect. Be sure it doesn't already contain lines numbered 0, 32766, or 32767. Then merge the above lines with your BASIC program by ENTERing the lines from disk.

3. Change the *filename.ext* in line 32767 to whatever name you wish to use for the protected version of your program.

4. Make sure the password disk is inserted in the drive. Type GOTO 32766 and press RETURN. When the READY prompt reappears, the program is protected. Now it can be run only after the password has been successfully entered with Atari Password.

Some password-protected programs have what's known as a *back door*. This is a secret way to bypass the protection. Atari Password doesn't have a back door, but it does have a secret feature that lets you exit the password program to BASIC without erasing the program from memory. When the main menu is on the screen, press the S key and wait. After five or ten seconds, the buzzer sounds. Then press the B key. You'll find yourself in BASIC with the password program intact.

How It Works

Here's a breakdown of Atari Password:

Lines	
1–6	These are DATA statements to refill the variable name table.
60–110	Initialization. Note the variables TRUE and FALSE which assign values to Boolean variables.
120	Contains the password. See explanation below for lines 6000–6080.
500	Checks to see whether the program has been previously run. If not, it checks for password. The memory address (1612) and code value (126) in this line can be changed to any free location and value that you would like, but be sure to reflect your change in line 5000 also.
510	The program reaches here only if the correct password has been entered. If so, it jumps to the main menu.
1000–1160	The main loop of the password-checking routine.
1200–1220	Backs up one space if SELECT is pressed while a password is being entered.
1300	Erases the entire input line if OPTION is pressed while a password is being entered.
1500–1510	Checks for the correct password if START is pressed.
1600–1650	Sounds alarm and displays the LOCKOUT message until START is pressed if the incorrect password is entered.
1999	Returns from the password-checking routine.
2000–2010	Fills the screen with inverse spaces.
2500–2600	Gets the actual password code from C$ and puts it in CODE$.
3000–3210	Main menu section.
6000–6080	This routine changes the password code. It does this by opening Atari Password on disk for read and write and searching for the occurrence of the two Z's that can be seen in line 120. When it finds this flag, it writes the new password code to disk. Something to note is the #16 in line 6005; this suppresses the question mark which is the normal INPUT prompt.
7000–7460	This routine calls the disk directory. It prints the directory on the screen along with an arrow-shaped pointer that can be moved to the desired filename. RETURN runs the selected program. If the entire directory cannot fit on the screen, the message <MORE> appears. Press START to see the rest of the directory or SELECT to go back to the main menu. This routine can be removed and used in your own programs, but remember to DIMension the variables DIR$, ENT$, and PR$, and use a GRAPHICS 0 statement, because the routine uses LOCATE to read the filename from the screen.
9000–9030	Restores the program to BASIC with the listing intact. This is done by refilling the variable name table with its original values, which are stored as DATA statements. Then it POKEs the correct length into line 9510. For this reason, you shouldn't change any of the variables in this program.
9500–9510	This short routine creates the protected version of the password program on disk.

For instructions on entering these listings, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!

Program 1: Atari Password

```
FD 1 DATA 84,82,85,197,70,65
     ,76,83,197,83,84,65,82,
     212,83,69,76,69,67,212,
     79
PC 2 DATA 80,84,73,79,206,84
     ,82,89,164,67,79,68,69,
     164,67,164,68,73,82,164
MM 3 DATA 69,78,84,164,80,82
     ,164,208,75,69,89,80,82
     ,69,83,83,69,196,67,79
FF 4 DATA 78,83,79,204,193,2
     01,217,78,67,164,68,79,
```

```
    78,197,68,77,65,216,77,
    79
IG 5 DATA 82,197,68,84,79,20
    8,66,79,212,216,68,69,2
    04,79,217,79,216,214,88
    ,67
JP 6 DATA 200,69,216,88,177,
    83,212,76,201,86,65,82,
    201,202
JN 60 OPEN #1,4,0,"K:"
BJ 70 GRAPHICS 0:POKE 82,2
AA 80 POKE 580,1:POKE 16,64:
    POKE 53774,64
LB 90 SETCOLOR 4,10,0:SETCOL
    OR 2,10,0
PJ 100 TRUE=(1=1):FALSE=(1=0
    ):START=6:SELECT=5:OP
    TION=3
CD 110 DIM TRY$(30),CODE$(30
    ),C$(32),DIR$(1235),E
    NT$(17),PR$(14),NC$(3
    0)
MH 120 C$="ZZENTER**********
    ****************"
GA 500 IF PEEK(1612)<>126 TH
    EN GOSUB 2500:GOSUB 1
    000:POKE 1612,126
JC 510 GOTO 3000
IA 997 END
PN 999 REM GET PASSWORD CODE
LH 1000 POKE 752,1:GOSUB 200
    0:POSITION 2,2:? "[GO
    DE]";:POSITION 7,2
FN 1010 P=0:TRY$=""
NP 1040 P=P+1:KEYPRESSED=FAL
    SE
DM 1045 IF P=30 THEN 1300
CL 1050 IF PEEK(764)<>255 AN
    D PEEK(764)<>154 THE
    N KEYPRESSED=TRUE
FC 1060 CONSOL=PEEK(53279)
OJ 1065 IF CONSOL=START THEN
     1500
BE 1070 IF CONSOL=SELECT THE
    N 1200
DD 1075 IF CONSOL=OPTION THE
    N 1300
DG 1100 IF  NOT KEYPRESSED T
    HEN 1050
GE 1110 GET #1,A
FJ 1130 TRY$(P)=CHR$(A)
HP 1140 PRINT "■";
MG 1160 GOTO 1040
DE 1198 REM BACKSPACE
HK 1200 IF P<=1 THEN GOTO 10
    50
EP 1210 P=P-1:TRY$(P)="":? "
    {8ELL}{LEFT}■{LEFT}";
ME 1220 GOTO 1050
EH 1298 REM CLEARLINE
EG 1300 POSITION 7,2:FOR I=1
     TO P:? "■";:NEXT I:
    TRY$="":POSITION 7,2
    :P=1:GOTO 1050
OK 1498 REM CHECK ENTERED CO
    DE
FK 1500 PRINT "{ESC}
    {CLR TAB}";:FOR I=1
    TO 200:NEXT I
CM 1505 IF TRY$<>CODE$ THEN
    1600
NM 1510 GOTO 1999
IG 1598 REM NO GOOD
AM 1600 GOSUB 2000:POSITION
    16,12:? "[illegible]"
HF 1610 SOUND 1,50,10,10:POK
    E 755,2:FOR I=1 TO 5
    0
JN 1620 IF PEEK(53279)<>STAR
    T THEN NEXT I
KB 1630 SOUND 1,100,10,10:PO
    KE 755,0:FOR I=1 TO
    50
NE 1640 IF PEEK(53279)<>STAR
    T THEN NEXT I:GOTO 1
    610
FP 1650 POP :SOUND 1,0,0,0:P
    OKE 755,2:GOTO 1000
LM 1999 RETURN
HH 2000 ? "{CLEAR}";:FOR Y=1
     TO 23:? "
    {38 [illegible]}";:NEXT Y
KD 2010 RETURN
IH 2500 P=3
HN 2510 IF C$(P,P)="*" THEN
    2600
MJ 2520 CODE$(P-2)=C$(P,P)
AD 2530 P=P+1
MH 2540 GOTO 2510
KI 2600 RETURN
PF 2998 REM MENU
GA 3000 PRINT "{CLEAR}{38 N}
    ";
MK 3010 PRINT "{18 SPACES}ME
    NU"
LP 3020 PRINT "{38 M}";
AI 3040 PRINT " 1 EXIT TO BA
    SIC"
PP 3050 PRINT " 2 CHANGE PAS
    SWORD CODE"
KN 3060 PRINT " 3 DISK DIREC
    TORY"
OK 3070 PRINT "{38 N}";
GM 3080 GET #1,A
BA 3090 IF A=ASC("S") THEN G
    OSUB 9000
LN 3100 IF A=ASC("8") AND LI
    =-999 THEN GRAPHICS
    0:END
LO 3200 IF A<49 OR A>51 THEN
     3080
CA 3210 ON A-48 GOTO 5000,60
    00,7000
FL 5000 GRAPHICS 0:POKE 580,
    0:PDKE 1612,0:NEW
MB 5999 REM CHANGE CODE WORD
OP 6000 PDKE 752,0
NF 6005 ? "{CLEAR}ENTER NEW
    CODE PASSWORD(S)";:I
    NPUT #16,NC$
FJ 6010 OPEN #2,12,0,"D:AUTO
    RUN.BAS"
AG 6020 IF LEN(NC$)>29 THEN
    NC$(29)=""
GL 6030 FOR I=LEN(NC$)+1 TO
    30:NC$(LEN(NC$)+1)="
    *":NEXT I
JI 6040 GET #2,A:IF A=90 THE
    N GET #2,A:IF A=90 T
    HEN 6060
MO 6050 GOTO 6040
NC 6060 FOR I=1 TO 30:PUT #2
    ,ASC(NC$(I,I)):NEXT
    I
JI 6070 CLOSE #2
PM 6075 PDKE 752,1
MK 6080 GOTO 3000
KA 6999 REM DIRECTORY ROUTIN
    E
HP 7000 OPEN #5,6,0,"D:*.*"
KI 7010 DIR$(1)="  ":DIR$(123
    5)="  ":DIR$(2)=DIR$
JN 7020 DONE=FALSE
PE 7030 POKE 752,1
II 7040 P=0
AF 7050 P=P+1
OM 7060 INPUT #5,ENT$
MN 7070 IF ENT$(5,8)="FREE"
    THEN DONE=TRUE
NM 7080 IF DONE THEN 7110
PB 7090 DIR$(P*19-18,P*19)=E
    NT$
MN 7100 GOTO 7050
II 7110 CLOSE #5:DMAX=P
BB 7120 P=1:MORE=FALSE:IF DM
    AX>40 THEN MORE=TRUE
FN 7130 DTOP=40:IF  NOT MORE
     THEN OTOP=DMAX
GM 7140 POKE 82,2:POKE 83,39
    :PRINT "{CLEAR}
    {13 [illegible]}
    {15 [illegible]}"
JL 7150 PRINT DIR$(P,DTOP*19
    );:IF OTOP/2<>INT(OT
    OP/2) THEN PRINT
DA 7160 BOT=PEEK(84)-1
KA 7170 ? " {36 R}";:IF MORE
     THEN POSITION 17,22
    :PRINT "<MORE>";
BK 7180 X=1:Y=1:POSITION X,Y
    :? "{ESC}{RIGHT}"
HI 7190 KEYPRESSED=FALSE
DB 7210 IF PEEK(764)<>255 TH
    EN KEYPRESSED=TRUE
FB 7220 CONSOL=PEEK(53279)
MB 7230 IF CONSOL=START AND
    MORE AND P<>1 THEN 7
    120
KC 7240 IF CONSOL=START AND
    MORE THEN P=(DTOP+1)
    *19+1:DTOP=DMAX:GOTO
     7140
BD 7245 IF CONSOL=SELECT THE
    N 3000
FL 7250 IF KEYPRESSED THEN 7
    270
NC 7260 GOTO 7210
IE 7270 DY=Y:DX=X
EN 72B0 GET #1,V:IF V=45 THE
    N Y=Y-1:IF Y<1 THEN
    Y=1
ME 7290 IF V=61 THEN Y=Y+1:I
    F Y>BDT THEN Y=BOT
HJ 7300 IF V=43 OR V=42 THEN
     XCH=XCH+1:IF XCH/2=
    INT(XCH/2) THEN X=1:
    GOTO 7320
BE 7310 IF V=43 OR V=42 THEN
     X=20
IH 7320 IF V=155 THEN 7350
PN 7330 POSITION DX,DY:? " "
    :POSITION X,Y:? "
    {ESC}{RIGHT}"
NI 7340 GOTO 7190
PP 7350 EX=0:PR$="D:":FOR X1
    =X+3 TO X+13
ND 7360 LOCATE X1,Y,V:IF V=3
    2 THEN 7400
BP 7370 PR$(LEN(PR$)+1)=CHR$
    (V)
AF 73B0 IF X1=X+10 THEN 7400
NK 7390 GOTO 7430
NO 7400 IF EX=1 THEN EX=0:GO
    TO 7420
KL 7410 X1=X+11:LOCATE X1,Y,
    V:IF V<>32 THEN PR$(
    LEN(PR$)+1)=",":X1=X
    1-1:EX=1:GOTO 7430
NF 7420 GOTO 7440
JG 7430 NEXT X1
ML 7440 TRAP 7450:RUN PR$
EM 7450 POSITION 13,23:? "CA
    NNOT BE RUN";:GET #1
    ,A
ME 7460 GOTO 7120
LO 8990 REM
FN 8991 REM THIS ROUTINE IS
    NOT ON THE MENU
AO 8992 REM IT RESTORES THE
    PROGRAM TO BASIC WIT
    H
HP 8993 REM THE LISTING INTA
    CT
OO 8994 REM TO USE THIS OPTI
    ON PRESS S AT THE ME
    NU
```

```
KG 8995 REM AND THEN PRESS 8
        WHEN YOU HEAR THE 8
        UZZER
MA 9000 ST=PEEK(136)+PEEK(13
        7)*256
BH 9010 LI=PEEK(ST)+PEEK(ST+
        1)*256:IF LI<>9510 T
        HEN ST=ST+PEEK(ST+2)
        :GOTO 9010
MH 9020 IF PEEK(ST+2)=0 THEN
        POKE ST+2,72:RESTOR
        E 1
DI 9030 FOR I=1 TO 115:LI=PE
        EK(130)+PEEK(131)*25
        6+(I-1):READ VARI:PO
        KE LI,VARI:NEXT I:LI
        =-999:? "{BELL}":RET
        URN
DI 9498 REM PROTECT SAVE ROU
        TINE
DG 9500 FOR VARI=PEEK(130)+P
        EEK(131)*256 TO PEEK
        (132)+PEEK(133)*256:
        POKE VARI,155:NEXT V
        ARI
EP 9510 POKE PEEK(138)+PEEK(
        139)*256+2,0:SAVE "D
        :AUTORUN.BAS":NEW
```

Program 2: Autoboot Maker

```
PA 10 OPEN #1,8,0,"D:AUTORUN
      .SYS"
PN 20 TRAP 40
BJ 30 READ A:PUT #1,A:CHK=CH
      K+A:GOTO 30
GJ 40 IF CHK<>10833 THEN ? "
      Error in DATA statemen
      ts!":END
PE 50 ? "AUTORUN.SYS file ha
      s been written."
CF 1000 DATA 255,255,0,6,109
        ,6
IN 1010 DATA 169,5,141,197,2
        ,133
GJ 1020 DATA 84,169,49,141,6
        8,3
GI 1030 DATA 169,6,141,69,3,
        169
CF 1040 DATA 0,141,73,3,169,
        61
IG 1050 DATA 141,72,3,169,11
        ,141
PI 1060 DATA 66,3,162,0,32,8
        6
ME 1070 DATA 228,169,0,133,8
        4,133
GE 1080 DATA 85,169,13,141,7
        4,3
DP 1090 DATA 96,71,82,46,49,
        43
DF 1100 DATA 49,54,58,63,35,
        54
FG 1110 DATA 59,34,32,32,32,
        207
FB 1120 DATA 206,197,160,205
        ,207,205
FN 1130 DATA 197,206,212,174
        ,174,174
ED 1140 DATA 34,58,80,79,75,
        69
CD 1150 DATA 32,56,52,50,44,
        49
DN 1160 DATA 50,58,82,85,78,
        34
ED 1170 DATA 68,58,65,85,84,
        79
EH 1180 DATA 82,85,78,46,66,
        65
DK 1190 DATA 83,34,226,2,227
        ,2,0,6
```

ProDOS File Converter For Apple *SpeedScript*

Leh-Wen Yau

This program speeds up conversions between ASCII text files and files compatible with the SpeedScript 3.0 *word processor, which was published in the June 1985 issue of* COMPUTE! *and is also available in book form* (SpeedScript: The Word Processor for Apple Personal Computers, *COMPUTE! Books). The new converter program works on all Apple IIe and IIc computers with ProDOS.*

Apple *SpeedScript* 3.0 (COMPUTE!, June 1985) is such a powerful word processor that many people, including myself, rely on it heavily for their word processing needs. I use the ProDOS version because it handles larger documents than the DOS 3.3 version. Written entirely in machine language, *SpeedScript* is efficient and fast.

However, the same cannot be said for the "*SpeedScript* File Converter" program, which lets you convert ASCII text files into *SpeedScript* files and vice versa. It's written in Applesoft BASIC and takes quite a few minutes to convert documents of any substantial size. Fortunately, it's not difficult to speed up the *SpeedScript* File Converter with the help of a short machine language routine. The modified File Converter completes its job within seconds, no matter how large a document is.

To see for yourself, type in and save Program 1 at the end of this article. When you run this program, it writes the machine language routine to disk under the filename SS.CONVERT. (Because Program 1 creates a file named SS.CONVERT, you *must not* use this name for Program 1 itself when you save it to disk.) You don't need to run Program 1 every time you want to use File Converter, just once to write the SS.CONVERT routine for File Converter to use. Then type in Program 2, the complete new version of the File Converter.

If you already have the old File Converter on disk, you can save some typing by modifying it rather than entering Program 2. Just follow these steps:

1. Type in and save Program 1, then run it to create the disk file named SS.CONVERT.
2. Delete lines 150 through 180 from the old File Converter.
3. Delete lines 240 through 260 from the old File Converter.
4. Add the following three lines to the Converter:

```
1 PRINT CHR$(4);"BLOAD
  SS.CONVERT"
150 CALL 768,8192,L-1,1
240 CALL 768,8192,L-1,0
```

5. Save the modified Converter on the same disk as the SS.CONVERT file.

When you run the modified Converter, it BLOADs the machine language routine from disk. It works just the same as the old File Converter in all other respects.

For instructions on entering these listings, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!

Program 1: Speed-Up Routine Generator For File Converter

```
8A 10 FOR I = 768 TO I + 145: RE
      AD A: POKE I,A: NEXT
94 20 PRINT CHR$ (4);"BSAVE SS.C
      ONVERT,A$300,L$92"
EE 30 END
31 100 DATA 32,183,0,201,44,208,
      3,32,190,222,32
FB 110 DATA 183,221,32,82,231,16
      5,80,133,250,165,81
3F 120 DATA 133,251,32,190,222,3
      2,183,221,32,82,231
1F 130 DATA 165,80,133,252,165,8
      1,133,253,32,190,222
BA 140 DATA 32,248,230,224,1,208
      ,46,160,0,165,251
81 150 DATA 197,253,208,6,165,25
      0,197,252,240,31,177
B9 160 DATA 250,201,60,208,6,169
      ,13,145,250,208,4
D3 170 DATA 41,127,145,250,24,16
      5,250,105,1,133,250
28 180 DATA 165,251,105,0,133,25
      1,208,213,96,224,0
D6 190 DATA 208,251,160,0,165,25
      1,197,253,208,6,165
60 200 DATA 250,197,252,240,237,
      177,250,201,13,208,6
DB 210 DATA 169,60,145,250,208,4
      ,9,128,145,250,24
7B 220 DATA 165,250,105,1,133,25
      0,165,251,105,0,133
A4 230 DATA 251,208,213
```

Program 2: *SpeedScript* File Converter For ProDOS

```
E7  1 PRINT CHR$(4);"BLOAD SS.CON
      VERT
4A 10 HOME
52 20 D$ = CHR$ (4)
25 40 PRINT "DO YOU WANT TO:"
A6 50 PRINT " (1) MAKE A SPEEDSC
      RIPT FILE INTO A TEXT
      FILE"
AE 60 PRINT " (2) MAKE A TEXT FI
      LE INTO A SPEEDSCRIPT
      FILE"
67 70 GET A$:A = VAL (A$)
47 80 IF A < > 1 AND A < > 2 THE
      N 70
65 90 ON A GOTO 100,200
53 100 PRINT "ENTER SPEEDSCRIPT
      FILE NAME": INPUT ":";A$
89 110 PRINT "ENTER TEXT FILE NA
      ME TO CREATE": INPUT ":";
      B$
7E 120 PRINT D$;"BLOAD ";A$;",A$
      2000"
A4 125 L = PEEK (48859) + PEEK (
      48860) * 256 + 8192
28 150 CALL 768,8192,L - 1,1
C9 190 PRINT D$;"CREATE ";B$;",T
      TXT"
F5 195 PRINT D$;"BSAVE ";B$;",A$
      2000,E";L - 1;",TTXT"
83 196 END
6D 200 PRINT "ENTER TEXT FILE NA
      ME": INPUT ":";B$
B6 210 INPUT "ENTER SPEEDSCRIPT
      FILE NAME TO CREATE   :";
      A$
25 220 PRINT CHR$ (4);"BLOAD ";B
      $;",A$2000,TTXT"
93 230 L = PEEK (48859) + PEEK (
      48860) * 256 + 8192
17 240 CALL 768,8192,L - 1,0
1A 245 IF PEEK (I) = 141 THEN PO
      KE I,60
4A 295 PRINT D$;"BSAVE ";A$;",A8
      192,E";L - 1
B4 296 END
```

HOTWARE: Software Best Sellers

This Month	Last Month	Title	Publisher	Remarks	Systems: Apple	Atari	Commodore	IBM	Macintosh
Entertainment									
1.	1.	*Ultima IV*	Origin Systems, Inc.	Fantasy game	•	•	•		
2.		*Gato*	Spectrum Holobyte	Submarine simulation	•		•	•	•
3.		*Hardball*	Accolade	Baseball game	•		•		
4.		*Bard's Tale*	Electronic Arts	Fantasy/role-playing game	•		•		
5.		*Flight Simulator*	Microsoft	Aircraft simulation				•	
Education									
1.	2.	*Math Blaster!*	Davidson	Introductory math program, ages 6-12	•	•	•	•	
2.	1.	*Typing Tutor III*	Simon & Schuster	Typing instruction program	•		•	•	•
3.	4	*New Improved MasterType*	Scarborough	Typing instruction program	•	•	•	•	•
4.	3	*Music Construction Set*	Electronic Arts	Music composition program	•	•	•		
5.		*Homework Helper: Math Word Problems*	Spinnaker	Math tutorial, high school level	•	•	•	•	
Home Management									
1.	1	*Print Shop*	Broderbund	Do-it-yourself print shop	•	•	•		
2.	2.	*The Newsroom*	Springboard	Do-it-yourself newspaper	•		•	•	
3.	5.	*Swiftax*	Timeworks	Tax preparation program	•		•	•	
4.	4.	*Print Shop Graphics Library*	Broderbund	100 additional graphics	•	•	•		
5.		*Paperback Writer*	Digital Solutions	Word processing program			•		

The Beginners Page

Tom R. Halfhill, Editor

More String Arithmetic

We've seen how to slice pieces out of character strings with LEFT$, RIGHT$, and MID$ ("The Beginner's Page," March and April 1986), and even how to use mathematical operators to compare the hidden number values in strings (May 1986). But "string arithmetic" doesn't stop there. BASIC also lets you *add* two or more strings to make an even longer string.

In Microsoft BASIC, this is a snap. (Computers with Microsoft or Microsoft-style BASICs include all Commodores, the Amiga, Apple II series, the Macintosh, IBM PC/PCjr, TRS-80, and Atari ST—but *not* the Atari 400/800, XL, or XE, although Microsoft BASIC is available as an option.) Here's an example:

```
10 A$="UNITED WE STAND; "
20 B$="DIVIDED WE FALL."
30 C$=A$+B$
40 PRINT A$
50 PRINT B$
60 PRINT C$
```

The result is:

```
UNITED WE STAND;
DIVIDED WE FALL.
UNITED WE STAND; DIVIDED WE
  FALL.
```

By adding A$+B$ into a new string variable, C$, we've preserved the original values of A$ and B$. But if this isn't a consideration, you can simply reuse one of the variables:

```
10 A$="UNITED WE STAND; "
20 B$="DIVIDED WE FALL."
30 A$=A$+B$
40 PRINT A$
50 PRINT B$
```

The result is:

```
UNITED WE STAND; DIVIDED WE
  FALL.
DIVIDED WE FALL.
```

As you can see, string addition is virtually identical to regular addition: The sum is the whole of the parts. In computerese, the fancy name for this is *string concatenation*. To impress people, try dropping this term into a conversation at your next user group meeting.

Although TI BASIC and Extended BASIC for the TI-99/4A are different in many respects from the other versions mentioned above, string concatenation is handled in a similar fashion. The only difference is that the concatenation operator is the & symbol instead of the + symbol. Any of the examples above can be used on the TI by substituting & wherever + appears.

The Fine Print

There's only one string attached when it comes to attaching strings: You have to be careful not to exceed the length limit for strings in your particular version of BASIC. You can hit this limit pretty fast because most Microsoft BASICs don't allow strings longer than 255 characters. An exception is Amiga BASIC, which allows strings up to 32,767 characters long.

Atari BASIC for the 400/800, XL, and XE computers also allows strings at least 32,000 characters long. As a matter of fact, on an Atari, you'll most likely run out of memory before you hit the length limit on strings. In effect, Atari BASIC lets you set your own length limits. Before using any string variable in an Atari BASIC program, you have to declare its maximum length with a DIM statement. For instance, DIM A$(100) means that A$ can hold up to 100 characters. Since a DIM statement forces Atari BASIC to immediately set aside the specified amount of memory for the string variable, the memory is protected. Nothing else, not even the BASIC program itself, can overwrite it. Many Atari programs take advantage of this megastring feature to reserve huge blocks of memory for holding data files and the like.

Nothing comes without a price, however, and one price you pay in Atari BASIC is that string concatenation is a little clumsier to write than it is in Microsoft-style BASICs. You can't simply add two strings together with the plus sign. Instead, it requires something like this:

```
10 DIM A$(50),B$(50),C$(100)
20 A$="UNITED WE STAND; "
30 B$="DIVIDED WE FALL."
40 C$(LEN(C$)+1)=A$
50 C$(LEN(C$)+1)=B$
60 PRINT A$
70 PRINT B$
80 PRINT C$
```

The result is:

```
UNITED WE STAND;
DIVIDED WE FALL.
UNITED WE STAND; DIVIDED WE
  FALL.
```

It takes two statements (lines 40 and 50) to accomplish the equivalent of C$=A$+B$ in Microsoft BASIC. Essentially, what these lines do is use the LEN function to say, "Starting at the last character in C$ plus one, append the contents of A$; then, starting at the last character in the new C$ plus one, append the contents of B$." Although not as readable or as easy to use as C$=A$+B$, the result is exactly the same.

If you're not interested in preserving the original contents of A$ and B$, it is possible to concatenate in one line. Substitute this statement and delete lines 50 and 80:

```
40 A$(LEN(A$)+1)=B$
```

The result is:

```
UNITED WE STAND; DIVIDED WE
  FALL.
DIVIDED WE FALL.
```

In either Microsoft or Atari BASIC, there's no such thing as string subtraction, multiplication, or division with the −, *, and / signs. Instead, you have to simulate these operations by slicing up the string with LEFT$, RIGHT$, MID$, and the other methods shown in the past few columns. ©

Printers For The Amiga

A printer is one of those optional but essential add-ons for your computer. It lets you reap something tangible from your word processor, terminal program, spreadsheet, or drawing program. True, you can use these tools to prepare files which you can transmit via modem directly to other computers. But hardcopy—type on paper—is still the only universally acceptable form of nonverbal communication.

Commodore doesn't sell an offical Amiga printer yet.Instead, you're free to attach practically any serial or parallel printer. The Amiga sports an RS-232 serial port as well as a standard parallel printer port. All it takes is the right cable to link the Amiga with almost any printer.

The commonly available IBM printer cables appear similar to Amiga cables, except the end of the cable that plugs into the IBM is a DB-25 male connector and the Amiga port is also a DB-25 male connector. Since printer cables specifically for the Amiga can be difficult to obtain, you might be tempted to use a "gender-changer" (a box or cable with a male connector on one end and a female connector on the other) to connect the IBM cable to your Amiga. *Don't do this.* Such an arrangement could damage your Amiga or your printer, or both.

The Amiga parallel port does not use exactly the same pin assignments as the IBM port. (Refer to page 7-13 of the *Introduction to Amiga* manual for a pinout chart.) Even more important, pin 23 on the Amiga parallel port is a +5-volt power supply, while pin 23 on an IBM-type printer cable may be connected to voltage ground. If the cable carries this voltage, and if the printer connector has a grounded pin at that position, the power supply in your Amiga may be damaged.

If you have a serial (RS-232) printer connected via the Amiga's serial port instead of the parallel port, a similar caution applies: Pins 14, 21, and 23 on the serial port carry power supply voltages. (Refer to page 7-12 of the *Introduction to Amiga* manual for a pinout chart.) Since these pins are often unused in devices like modems and printers, it *may* be safe to use IBM-type serial printer cables. Check the manual for your printer carefully to be sure that your particular model does not make any connection to these pins. Again, a gender-changer plug will be required to attach an IBM-style cable to the Amiga serial port. It's best to check with your dealer before using a suspect cable.

Printer Drivers

Once you've hooked up the hardware, you need to "attach" the printer to your software. Although every printer manufacturer uses different specifications for software control over printing features, the Amiga is capable of adapting to a variety of popular printers.

What complicates things is that every printer has its own unique set of codes, even for common effects such as underlining or boldfacing. For example, the Epson MX-80 uses the ASCII sequence 27-53 ("ESC-4") to turn on italics mode, and 27-54 ("ESC-5") to turn off italics. On the other hand, the Okimate 20, which is similar in many other ways, uses the sequence 27-37-71 (ESC-%G) to turn on italics, and 27-37-72 (ESC-%H) to turn italics off.

When an Amiga program wants to print italics, it can't just use the code for one printer model, because the program would be incompatible with other printers. Instead of sending the actual code for italics, then, Amiga programs send a symbolic code for italics. If you tell the Amiga which *printer driver* to use, the driver translates these symbolic codes into the actual codes for your printer.

Use Preferences to install your printer driver, following the instructions given in *Introduction to Amiga*, pages 7-6 to 7-11. Printer drivers currently exist for the Alphacom Alphapro 101 (no longer in production), Commodore CBM-MPS 1000, Epson FX-80, RX-80, HP Laserjet/Laserjet Plus, Brother HR-15XL, Diablo Advantage C-150, D25, 630, Qume LetterPro-20, and Okimate-20. If your printer is not on this list, try some of the drivers to see if they work with your printer. For example, the Juki 5510 dot-matrix printer is Epson JX-80–compatible, so you can use the Epson JX-80 printer driver.

If none of the drivers work, select the Custom printer driver. If you have the version 1.1 operating system upgrade, by default the Custom selection looks for a printer driver named Generic. The Generic driver works with any properly cabled printer by ignoring all special printer codes. If your printer won't respond to the codes used by any of the printers on the list, you can at least get a plain-vanilla text printout with the Generic driver.

Unfortunately, the Generic driver won't let you use any special printing effects such as underlining, boldface, italics, or bit-image graphics. You need a printer driver created especially for your brand of printer. Many people are working on drivers for unsupported printers, including one company that has developed a printer-driver builder that a nonprogrammer can use to design a new custom driver. Nevertheless, if you are looking for a printer for your Amiga, it's best to buy one that is compatible with one of the above printers.

Donald B. Trivette

Softstripping

What avid reader of COMPUTE! hasn't wished for a magic way to get the program listings off the pages of the magazine and into the computer?

The device in the photograph, called a Softstrip Reader by Cauzin Systems, does just that—almost. It can't read English, or even BASIC, but it can read any program, text, or data that has been encoded in the Softstrip format—a kind of universal product code Cauzin has developed for computers. The black-and-white strip you see here, which looks like something you rub with a coin to find out you've lost a contest, is in fact Lincoln's Gettysburg Address. All 276 words of it are contained in the six-inch strip.

The Softstrip system consists of both hardware and software. The hardware is a reader the size of a giant rolling pin that plugs into the computer's serial port just like a modem. It gets power from a small transformer that plugs into a wall outlet. A truck inside the D-shaped plastic case moves the read-head down the strip when the reader is activated.

The software is a program called *Cauzcomm*. On the IBM, you can run *Cauzcomm* by typing its name at the DOS prompt, or by installing it as a resident program which is called up at any time by pressing the Alt-R keys. Once running, *Cauzcomm* displays a simple four-item menu: Read, Help, Options, Quit. To read a Softstrip, you align the reader over the strip, using the black dot and the heavy line as guides, then select Read from the menu. In less than 20 seconds, *Cauzcomm* reads the text of the Gettysburg Address into a disk file named GETTY.TXT. It couldn't be easier or more straightforward.

Make Your Own, Too

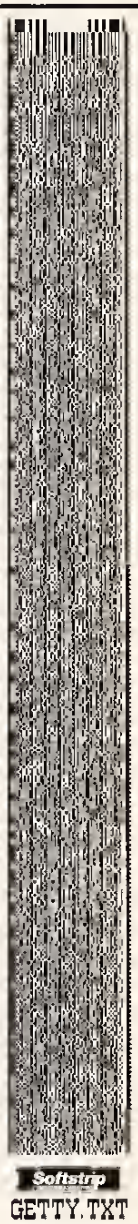
Examine the strip and you'll see the header markings at the top. The header indicates the filename and whether the strip is intended for an Apple or an IBM computer. It also tells the number of characters (bytes) in a horizontal line (typically four), the height of each line (typically 12/1000 inch), and the paper-to-ink contrast level. You'll see markings called the *checkerboard* running vertically down the left edge of a strip, and along the right edge, the *rack*. These denote each horizontal line and send alignment information to the reader. There's a parity mark at the end of each line for error detection.

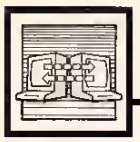
Cauzin System's Softstrip Reader is a new type of bar code reader that speeds up the entry of published program listings.

Cauzin sells an optional program that lets you make your own Softstrips with a dot-matrix printer. Or, for about $20, you can have Cauzin make a denser negative—up to 5,500 characters in a nine-inch strip—suitable for publication. Strips may be printed on almost any kind of paper, although lower densities are recommended for porous grades of paper. The strips can even be photocopied.

The reader comes with a booklet of 48 BASIC programs, and Cauzin plans to attract buyers by publishing programs in its advertisements in many computer magazines. The reader costs $200 and is available for the IBM PC/PCjr, the Apple II series, and the Macintosh. For more information, write to Cauzin Systems, 835 South Main Street, Waterbury, CT 06706.

Will Softstrips ever become a standard part of published program listings? Right now, it's a chicken-and-egg situation: Magazines and books may not print Softstrips until a sizeable number of their readers own the devices, while people interested in Softstrips may hold off buying a reader until Softstripped listings become more common. If you want to express an opinion on this topic, write to COMPUTE!, P.O. Box 5406, Greensboro, NC 27403.©





Telecomputing Today

Arlan R Levitan

The Price Of Telecomputing

Folks, I've got a confession to make. This page came very close to being blank this month. I was far beyond my habitual fashionably late, two-weeks-behind-deadline mode of operation. Even Bill Wilkinson had sent in his column for this month, an event usually reserved for coinciding lunar and solar eclipses occurring on February 29. What was wrong?

I was, dear friends, becalmed in the telecomputing doldrums. Me, of all people, contemplating the light-emitting diodes of my collection of modems for hours on end, at a loss for words! What was the cause of this strange malaise? After all, I had been using three new commercial information services over the last month. And a rundown on one alone is usually fodder enough for a good column.

Both General Electric's Genie and ViewTron's ViewData are being heavily promoted as the latest and greatest information services for computer hobbyists. I'll be glad to give credit where it is due—both GE and ViewTron have created relatively smooth systems with decent user interfaces. But I find that both are lacking in originality. Both services stick to what is by now the standard formula of special interest groups (SIGs), online conferencing and magazines, public domain libraries for downloading, shopping services, and games.

Then there's BIX (BYTE Information eXchange). BIX makes no pretenses about being everything to everybody. It is first and foremost a message-based conferencing system. While BIX's scope may be limited and its ease of use leaves something to be desired, the quality of its user base is the big attraction. BIX users tend to be technophiles. If you're having trouble debugging a LISP program or want to add an RS-232-controlled Veg-A-Matic to your system, you can probably find help on BIX.

Time Is Money

So why am I grousing? I'm becoming concerned with the pricing of time on the commercial services intended for home users.

I was one of the early users of the online services and I remember what we paid for nonprime time way back when: Two and a half bucks an hour was the going rate for 300 bps speed, and if you waited until the wee tiny hours of the morning, you could even run 1200 bps for under five bills. Most early users also recall the promises of even lower rates once the user base was expanded. Instead, the hourly access charges for nonprime time use have steadily risen.

Yes, it's true that the rates for daytime access have fallen. And it's true that many more functions have been added to the various services. And, yes, the cost of staffing has risen over time. However, the cost of computing power and data storage has dramatically fallen during the same period.

In the late 1970s, Scott Adams of Adventure International was once asked how he priced his popular series of adventure games. Adams replied that he used the first-run movie method of pricing. His basic premise was that consumers should get some hours of use from any software purchase and should pay no more for that use than the hourly cost of attending a first-run movie. That works out to about two to three bucks an hour at today's prices.

If you accept that formula, it's not hard to come to the conclusion that most of today's information services (even the "bargain" services) are expensive—especially when compared to today's hardware prices. The owner of a $130 Atari 130XE or Commodore 64 can easily fork out $40 to $80 a month for using an information service a couple of hours a week.

Drinking From The Well

One hopeful glimmer of sunshine is The Well, a project of Stewart Brand and his cadre of Whole Earth Software Catalog counterculture techno-renegades. The Well is a low-cost ($2/hour) conferencing system for the San Francisco Bay area. The system runs on a VAX minicomputer with a capacity of 40 phone ports at the offices of the Whole Earth Catalog and Whole Earth Review in Sausalito. The service was codeveloped with NETI (Network Technologies, International) of Ann Arbor, Michigan.

I've accessed The Well via PC Pursuit and have found it to be a conferencing system of extremely high quality. The Well's biggest problem is the relatively low number of users the system can support at one time.

Let's hope we'll see a proliferation of systems like The Well in the future. If Brand and his cohorts are willing to share the system software with other groups of like-minded enthusiasts, that may indeed come to pass. Since The Well's software is Unix-based, it can likely be ported to a Unix-capable mainframe computer 10 to 20 times the size of The Well's VAX. Such systems could support 300 to 400 users at a crack and are readily available on the used market at bargain prices.

I predict that someone out there will make it happen within the next two years. Keep your eyes and ears open and the bucket ready...you may be dipping into a Well soon. ©

Bill Wilkinson

ST System Software, Inside Out

Okay, you've got your shiny new ST computer plugged in and running. You can use the mouse to select programs, copy files, and format disks. It's fun, and it certainly is easier to learn than figuring out what

```
COPY B:\SYSTEM\MSGS.TXT/A=A:
  SPCL*.MS?
```

is supposed to mean. (That's a real and possible IBM PC command.) But how did this system get built? Glad you asked.

Collectively, the software built into the Atari is called TOS (Tramiel Operating System). When the 520ST was first shipped, TOS was delivered on a disk. If you're still using the disk-based TOS, stop now. Go out and buy the ROM (Read Only Memory) version of TOS. It should cost no more than $25 or so. Installation is not too difficult, though if you have as many left thumbs as I do, you might be advised to find a dealer or service center to install the chips for you (maybe $20 to $30 extra).

TOS in ROM is actually composed of six separate pieces. Usually, we lump these pieces into two groups of three: the graphics processing section and the underlying operating system. As we shall see, that operating system—a derivative of CP/M-68K—is very similar to MS-DOS and PC-DOS, which are both derivatives of CP/M.

BIOS, XBIOS, And GEMDOS

In one sense, we can say that the lowest level of the ST's operating system is the *BIOS* (Basic Input/Output System), a holdover from the earliest days of CP/M. At this level, we find routines for such basic tasks as sending a single character to a device, reading a disk sector (by sector number—a *very* dangerous practice), and so on. In CP/M, there was only one legitimate reason to call the BIOS directly: speed. With TOS, though, only the BIOS provides some of the facilities which even a moderately sophisticated program will need (admittedly, often because of bugs in the upper levels of the operating system).

On the ST, a BIOS call is implemented as a TRAP instruction in 68000 machine language. All the necessary parameters, including the BIOS call number, are passed onto the stack. If you aren't quite sure what we're talking about, don't worry about it. Virtually every programming language for the ST has some way to use these routines which mask the mechanics of TOS calls. It's a good thing, too, since some of those mechanics can get pretty hairy.

The next higher component of TOS is the *XBIOS* (eXtended BIOS). XBIOS supplies the Atari-unique routines needed to do such things as access the sound registers, screen hardware, and so on.

The third component of the operating system is called *GEMDOS* Graphics Environment Manager/Disk Operating System). Actually, this is a misnomer. The GEMDOS routines have nothing whatsoever to do with graphics. GEMDOS is essentially an MS-DOS or PC-DOS emulator. Want to open a file? Read a block of bytes? Get a character from the keyboard? Given the differences between the 68000 of the ST and the 8088 of the IBM PC, the similarities between GEMDOS calls and MS-DOS calls are almost scary.

GEM, VDI, And AES

Okay, enough about the underlying operating system. Let's take a look at the graphics systems which comprise GEM. The most familiar part is the *GEM desktop* which appears when you turn on your ST. But the desktop is not really a special program at all; it simply calls the lower-level routines. Again, there are three levels of graphics routines.

The lowest-level graphics, not officially part of GEM but merely one means of implementing it, are those called the *Line-A Routines*. This sounds cryptic, but it simply refers to the fact that certain machine instructions of the 68000 (including those of the form $A*xxx* hex—hence "line-A") are reserved and cause a special hardware trap into the OS. As you might expect, routines implemented in this fashion are of the most fundamental type: draw a line, plot a point, and so forth. Most are very fast.

The next level up in graphics is the *VDI* (Virtual Device Interface). In theory, VDI is capable of supporting several types of graphics devices in a uniform fashion. For example, you might use the same set of calls to draw a curve on a plotter or on the screen. Unfortunately, no such drivers are yet available (or, as far as I can tell, even in the works) for the ST. Still, the possibility exists.

VDI does all the actual graphics work on the ST. It draws simple rectangles, bordered ovals, and text in various styles, sizes, and colors. Someone who learns nothing on the ST except how to call VDI could still do some remarkable graphics work.

Finally, at the highest level, is *AES* (Applications Environment System). AES is what GEM uses to present you with that nice, pretty desktop, complete with menus, dialog boxes, alert boxes, windows, and icons. Perhaps more important to programmers, though, is the fact that AES allows us to use all the features of GEM in a relatively consistent, properly desktop-compatible manner. It is through this mechanism that even a lowly spreadsheet program can have drop-down menus, mouse-controlled windows, and all the rest of those impressive features. ©

Computers and Society

David D Thornburg, Associate Editor

Printers And Computers

Printer technology has advanced on so many fronts at once that it's hard to keep up to date. On one extreme is the continued push to ever-lower prices. The lowest-cost printers incorporate dot matrix mechanisms that either hammer ink onto paper through a ribbon, or that use thermal energy to transfer ink or induce color changes in special papers. These technologies have become so inexpensive that I recently saw a computer-controlled electronic typewriter that retailed for under $70.

Printers Smarter Than Their Computers

At the other extreme are the laser printers that combine xerographic copier technology with a computer-controlled laser beam to build up images on a photosensitive drum. The developed image is then transferred to a piece of paper at a resolution of 300 dots per inch. These printers are available at prices ranging from about $3,000 to $6,000 or so, and they often contain dedicated computers that can outperform the computer that is sending information to be printed. For example, the Apple LaserWriter (that I have connected to my Apple II as well as to my Macintosh) contains a 68000-based computer that can be programmed by the user through a Forth-like language called PostScript. The images created by this printer are exceptional in their quality.

It is interesting to note that, at both ends of the price spectrum, printers build images from an array of dots. The main difference between the extremes is in resolution, speed, and image quality. The market for various printers is sufficiently large that all kinds and prices of printers are enjoying a brisk business.

Considering the major impact that dot-matrix printer technologies have had on the personal computer industry, one wonders what will happen to the traditional printer that uses a daisy wheel or other mechanism to produce letter-quality type. These printers are still considered essential by most businesses, where low-resolution dot-matrix images are considered unacceptable. But letter-quality printers are comparatively expensive (at least the rugged ones are), and their price falls in the middle of the printer spectrum.

My concern for this technology is that it is being eroded from both ends. The low-cost printers are producing higher and higher quality images, and the laser printers are getting cheaper. Within the next few years the high-quality impact printer may become little more than a curiosity—used by people who, like me, prefer a fountain pen to a felt tip.

The difference between the dot-matrix and daisy-wheel printers is more than quality and price. The daisy-wheel printer is limited to printing text. Dot-matrix printers, on the other hand, can be used to prepare text or graphics, since both words and pictures can be formed from patterns of dots. This creative freedom of dot-matrix printers has other consequences. For example, if the resolution is high enough, text can be created in numerous styles (roman, italic, bold), type sizes, and fonts (character shapes).

Text Is Graphics

Text documents, as typesetters have known for centuries, are *graphics* documents as well. This realization is especially evident in laser printers, where the high resolution lets anyone do their own typesetting. Computer users who used to concern themselves with only spelling and grammar are now talking about *leading* (rhymes with *bedding* and refers to the blank space between lines of text), *points* (units of measurement equal to 1/72 of an inch), *intercharacter justification* (aligning columns of type), *kerning* (adjusting the spacing between two characters to be closer), *ligatures* (twin characters of type), and other terms that were rarely heard outside the walls of typesetting companies.

The most exciting aspect of low-priced laser printers is that small companies (and fortunate individuals) can be their own publishers. The economic justification for desktop publishing is easy to see. Suppose you are a software publisher who wants to create nice-looking manuals. The typesetting, proofreading, and editing of a 100-page manual can cost several thousand dollars and take several weeks. For a similar investment you can purchase a laser printer and, using documents written with your word processor, typeset the manual yourself in a day or two. The investment can pay for itself with the very first job.

You may think of the printer as a simple extension of the computer. It is far more than that—it is a tool that lets your creativity reach beyond the computer to touch others. Not a bad accomplishment for a mechanical contraption.

Dr. Thornburg's most recent product is Calliope, *a nonlinear idea processor for the Apple IIe, IIc, and Macintosh computers. He welcomes letters from readers and can be reached in care of* COMPUTE!. *He has just published* Unlocking Personal Creativity, *a book on creative problem-solving that he wrote and typeset himself using the Apple LaserWriter.* ©

The World Inside the Computer

Fred D'Ignozio, Associate Editor

A Multimedia Workstation For Teachers

One of the most exciting trends in low-cost computing is linking computers with other devices that record, edit, and play electronic media. For example:

- A MIDI (Musical Instrument Digital Interface) box lets you plug your computer into a variety of keyboard synthesizers, drum machines, guitars, and other instruments.
- A video camera lets you shoot images from the screen of your computer to use in your video presentations.
- A video digitizer allows you to shoot video images with your camcorder or video camera and transfer them to your computer.
- A SMPTE (Society of Motion Picture and Television Engineers) interface lets you synchronize your own music and sound effects with your videotapes.
- Scanners, graphics tablets, and graphics printers let you copy graphs, maps, diagrams, and artwork into your computer so they can be manipulated, labeled, and printed on paper or transparencies for overhead projector presentations.
- Graphics-design and animation programs can be used to create artwork and titles that can be copied with your video camera and edited into video presentations.

Ideal For Schools

All these devices can be assembled into a single multimedia workstation for under $3,000. This is a sum that most schools can afford, especially since only one or two workstations would satisfy a school's needs for the immediate future.

Who would use this workstation? Teachers often feel they've been overlooked by the computer revolution because most of the software and applications are intended for their students. A multimedia workstation would be different. Its primary purpose would be to help teachers prepare audiovisual materials for their classrooms. Student use might come later, but it would stem naturally out of the teachers' enthusiasm for using the workstation and their desire to share its capabilities with students.

Until the last year or two, only a TV station or a rich corporation or an ad agency or a major rock star could afford to create multimedia programming. The rest of us had to be content with doing all our communicating live, or via the printed page or audio tape.

Now, suddenly, things are changing. Machines, software, and techniques which once cost tens of thousands of dollars are becoming available for home and school computers. We now have the opportunity to communicate in several new mediums—and combinations of mediums—including videotape, graphics, music, sound effects, and professional-looking publications.

But a big question remains: Will we make the switch? Most of us are too accustomed to being media consumers rather than producers. Also, we may have great confidence in our ability to stand up in front of a group of youngsters and communicate with them verbally or with the printed word, but we are intimidated by the thought of creating our own movie, slideshow, or graphics presentation.

You Are Steven Spielberg

It's time we started learning. Electronic media is the wave of the future. We are surrounded with powerful electronic programming produced by people who want to sell us things: perfume, a new car, records, a new political candidate. It's time that teachers generated their own programming that communicates their special passions, enthusiasms, and pet subjects.

But most of us are novices in this area. How do we get started? A good way to begin learning how to be a media producer is to become a more critical media consumer. Switch on your TV, your record player, or your cassette player, and look and listen very carefully. Try to focus just on the sound—and on your reaction to the sound. Then turn off the sound and concentrate on the picture. What kinds of camera shots and special effects is a program using? Two good sources for quick courses in media production are commercials (slick and short) and MTV (unpolished and experimental).

Now it's your turn. I would like you to tell me what you'd most like to do. Pretend for a moment that you are Steven Spielberg, and you still have a fifth-grade class to teach in Little Rock, Arkansas, or Halifax, Nova Scotia. What ideas or subjects thrill you but have been difficult to get across in traditional ways? What areas in your curriculum are crucial for children to learn but for which you lack adequate materials? What are the special pet areas that *you* love to learn or teach that you'd like to share with your kids?

Please write me (care of COMPUTE!) and tell me what you'd like to teach using a multimedia workstation and how you would present it (with videotape, music, mixture of live-action shots, computer graphics, field trips—whatever!). And don't hold back. Be imaginative, creative, even far out. I want as many ideas as possible, since I'll be publishing them in an upcoming COMPUTE! column. ©

Programming the TI

C Regena

Printing A Schedule Of Events

The first peripheral I got for my TI was a printer. At first I used it mainly for program listings. Later, I discovered that adding a printer significantly increased the possible applications for the computer. All kinds of reports could be generated, lists sorted, and charts and graphs plotted. For some reason, if something was printed using a computer it looked more "official." Of course, a report is really only as good as whatever the programmer or computer user enters, but using your TI and a printer, you can make very impressive reports.

On the other hand, if you don't want something to look computer-generated, you can use a letter-quality printer. When the TI-99/4 first came out, only one printer was available for it (remember the TI thermal printer?). Next came a peripheral system that required the RS-232 interface which could be used with several different brands of printers. Now there are many more printers available and several kinds of interfaces or special cables, so there is no one standard way of using a printer. There are also several word processing programs available.

Printing In BASIC

In TI BASIC programming, the most common way to print something is to use the PRINT # statement (pronounced *print file*), which means print to a file or device. First, use OPEN # (*open file*) to define the printer for the computer. The manual for the interface or printer you use should have sample OPEN statements for your particular printer. When you're finished printing, use CLOSE # (*close file*). For example, here's what I need for my TI printer:

```
100 OPEN #1:"RS232.BA=600"
```

To print a message, for example:

```
110 PRINT #1:"HELLO"
```

Then, when finished, use:

```
900 CLOSE #1
```

The critical statement is the OPEN # statement, which must be tailored to your own printer configuration. The PRINT # and CLOSE # statements can be the same for any type of printer.

This month's program illustrates the use of a printer for creating a simple schedule of events or calendar of happenings. This program just gives the basic idea of how you can sort events by date and time. For example, you could use this program to list your activities for the summer. With my large family, I need to keep a written list of what's going on. To customize this program, add your own title and change the printing to fit your needs. The program as is simply lists the dates, times, and events in single spacing, but all kinds of special formats are possible, including a full, graphic calendar.

How It Works

The events are listed in DATA statements. For examples, I have used several events in lines 1240–1480. The last DATA statement should use 9999 for the date. The data for each event consists of the date, the time, and a description of the event. The date is a four-digit string. The first two numbers range from 01 to 12, representing the month. The last two numbers are the day of the month (01–31). The time is also a four-digit string. This is a number expressed as 24-hour time without a colon between the hours and minutes. Thus, 0800 is eight o'clock in the morning, and 1200 is twelve o'clock noon. Eight o'clock in the evening expressed in 24-hour time is 2000. For no specified time, use 0000.

This format helps in the sorting procedure. You may prefer to enter the data in a different way, then let the program convert to numbers for sorting. The numbers are converted to months, days, hours, and minutes during the printing procedure.

Lines 110–120 dimension variables DT$ (date and time) and EVENT$ for 50 events starting with base 1. M$ is dimensioned and will hold the names of the 12 months. Lines 200–240 define these month names in the M$ array.

The variable E is the number of an event. Line 270 READs from the data the date, time, and event. Line 280 checks to see whether all the data has been read. Line 300 increments E; then line 310 makes sure E is less than 51 for the subscript.

Lines 330–640 sort the events by date and time. I call this type of sort "maximum-minimum" because the first pass through the data finds the maximum and minimum items in the array and places them at the end points. Successive passes through the items move the ends inward and place the maximums and minimums at those positions.

Lines 740–1190 print the events in date order. Remember to put your own printer configuration in the OPEN statement in line 760. The variable DT$ is divided back into DATE$ and TIME$. The date is then separated so that a month name is printed with the day. The time is converted to the usual written format of hour:minute, and A.M., NOON, or P.M. is added.

If you don't have a printer, you can simply print the schedule on the screen. To control scrolling, PR is used as a variable to count how many lines have been printed on the screen. When the printing stops, press the space bar to continue the list. At the end of the list, press the space bar to get back to the menu screen.

If you wish to save typing effort, you can receive a copy of this program ("TI Calendar") by sending

a blank cassette or disk, a stamped, self-addressed mailer, and $3 to:

C. Regena
P.O. Box 1502
Cedar City, UT 84720

TI Calendar

```
100 REM  CALENDAR
110 OPTION BASE 1
120 DIM DT$(50),EVENT$(50)
130 DIM M$(12)
140 CALL CLEAR
150 PRINT TAB(6);"** CALENDAR **"
160 PRINT :::"ENTER DATES AND EVENTS IN"
170 PRINT :"DATA STATEMENTS."
180 PRINT ::"YOU MAY PRINT THE CALENDAR"
190 PRINT ::"ON THE SCREEN OR PRINTER.":::
200 FOR M=1 TO 12
210 READ M$(M)
220 NEXT M
230 DATA JAN,FEB,MAR,APR,MAY,JUN
240 DATA JUL,AUG,SEP,OCT,NOV,DEC
250 PRINT "...READING DATA..."
260 E=1
270 READ DATE$,TIME$,EVENT$(E)
280 IF DATE$="9999" THEN 320
290 DT$(E)=DATE$&TIME$
300 E=E+1
310 IF E<51 THEN 270
320 E=E-1
330 PRINT :"...SORTING..."
340 N=E
350 S=1
360 MN$=DT$(S)
370 IMIN=S
380 MX$=MN$
390 IMAX=S
400 FOR I=S TO N
410 IF DT$(I)<=MX$ THEN 440
420 MX$=DT$(I)
430 IMAX=I
440 IF DT$(I)>=MN$ THEN 470
450 MN$=DT$(I)
460 IMIN=I
470 NEXT I
480 IF IMIN<>N THEN 500
490 IMIN=IMAX
500 AA$=DT$(N)
510 BB$=EVENT$(N)
520 DT$(N)=DT$(IMAX)
530 EVENT$(N)=EVENT$(IMAX)
540 DT$(IMAX)=AA$
550 EVENT$(IMAX)=BB$
560 N=N-1
570 AA$=DT$(S)
580 BB$=EVENT$(S)
590 DT$(S)=DT$(IMIN)
600 EVENT$(S)=EVENT$(IMIN)
610 DT$(IMIN)=AA$
620 EVENT$(IMIN)=BB$
630 S=S+1
640 IF N>S THEN 360
650 PRINT :::"CHOOSE:"
660 PRINT :"1 PRINT ON SCREEN"
670 PRINT :"2 PRINT ON PRINTER"
680 PRINT :"3 END PROGRAM"
690 CALL KEY(0,K,S)
700 K=K-48
710 IF (K<1)+(K>3)THEN 690
720 CALL CLEAR
730 ON K GOTO 770,760,1490
740 REM  PRINTING
750 REM  PUT PRINTER CONFIGURATION HERE
760 OPEN #1:"RS232.BA=600"
770 FOR T=1 TO E
780 DATE$=SEG$(DT$(T),1,4)
790 M=VAL(SEG$(DATE$,1,2))
800 MON$=M$(M)
810 DAY$=MON$&" "&SEG$(DATE$,3,2)
820 TIME$=SEG$(DT$(T),5,4)
830 IF TIME$<>"0000" THEN 870
840 TIME$="      "
850 T$="     "
860 GOTO 1000
870 H=VAL(SEG$(TIME$,1,2))
880 IF H>=12 THEN 910
890 T$="A.M."
900 GOTO 970
910 IF H<>12 THEN 950
920 IF SEG$(TIME$,3,2)<>"00" THEN 960
930 T$="NOON"
940 GOTO 970
950 H=H-12
960 T$="P.M."
970 TIME$=STR$(H)&":"&SEG$(TIME$,3,2)
980 IF LEN(TIME$)>4 THEN 1000
990 TIME$=" "&TIME$
1000 TT$=TIME$&" "&T$
1010 IF K=2 THEN 1110
1020 PRINT :DAY$;"    ";TT$
1030 PRINT "  ";EVENT$(T)
1040 PR=PR+3
1050 IF PR<24 THEN 1120
1060 IF T=E THEN 1120
1070 CALL KEY(0,K,S)
1080 IF S<1 THEN 1070
1090 PR=0
1100 GOTO 1120
1110 PRINT #1:DAY$;"   ";TT$;"     ";EVENT$(T)
1120 NEXT T
1130 PR=0
1140 IF K<>2 THEN 1170
1150 CLOSE #1
1160 GOTO 650
1170 CALL KEY(0,K,S)
1180 IF S<1 THEN 1170
1190 GOTO 650
1200 REM  DATA FOR EVENTS
1210 REM  DATE,TIME,EVENT
1220 REM  DATE IS MMDD
1230 REM  TIME IS HHMM
1240 DATA 0415,0000,CINDY'S BIRTHDAY
1250 DATA 0415,1700,MAIL IN COME TAX
1260 DATA 0509,0000,RICHARD'S BIRTHDAY
1270 DATA 0510,0000,BOB'S BIRTHDAY
1280 DATA 0611,0000,CHAN'S BIRTHDAY
1290 DATA 0304,1200,SUSC VS BYU BASEBALL
1300 DATA 0305,1200,SUSC VS BYU BASEBALL
1310 DATA 0314,1300,SUSC VS UTAH BASEBALL
1320 DATA 0315,1300,SUSC VS UTAH BASEBALL
1330 DATA 0328,1300,SUSC VS MESA BASEBALL
1340 DATA 0329,1230,SUSC VS MESA BASEBALL
1350 DATA 0429,1300,SUSC BASEBALL
1360 DATA 0430,1300,SUSC BASEBALL
1370 DATA 0314,0715,SKI CLUB--RICK
1380 DATA 0225,0000,CINDY SKIING
1390 DATA 0328,1500,CHERY PARTY
1400 DATA 0222,1930,SUSC BASKETBALL
1410 DATA 0303,2000,SYMPHONY
1420 DATA 0330,0000,EASTER
1430 DATA 0526,0000,MEMORIAL DAY
1440 DATA 0607,0000,COMMENCEMENT
1450 DATA 0704,0000,INDEPENDENCE DAY
1460 DATA 0724,0000,PIONEER DAY
1470 DATA 0710,1000,SHAKESPEARE FESTIVAL
1480 DATA 9999,0000,ZZZ
1490 END
```

©

Attention Programmers

COMPUTE! mogozine is currently looking for quolity orticles on Commodore, Atori, Apple, ond IBM computers (including the Commodore Amigo and Atori ST). If you hove an interesting home opplicotion, educotionol program, progromming utility, or gome, submit it to COMPUTE!, P.O. Box 5406, Greensboro, NC 27403. Or write for o copy of our "Writer's Guidelines."

COMPUTE!

IBM Variable Snapshot

In line 1760 of this utility from the April issue (p. 81), the = before the value 65536! should instead be −.

Apple Switchbox

The + symbol in line 346 of this game from the March issue (p. 47) should instead be =. ©

INSIGHT: Atari

Bill Wilkinson

Atari Printer Trivia

This month's COMPUTE! is a printer issue, so I decided to break with (my) tradition and write a column on printers. Before we start, though, an erratum: My April column listed a program designed to "unify" a machine language file on disk. But when I sent the column to COMPUTE!, I accidentally included a couple of older versions of the program on the same disk. Guess which version got published? Anyway, COMPUTE! listed a corrected version in the article entitled "Custom Characters for Atari *SpeedScript*" in the May issue. (By coincidence, it happens that my program is needed to unify the COMPUTE! DISK version of *SpeedScript* before installing a custom character set.) On to the printers.

Number, Please

John Skruch at Atari gets credit for revealing this first tidbit. You are all aware that disk drives can be assigned device numbers (from D1: to D8:, though Atari drives can only go to D4:), but did you know that printers can have numbers, also? If you have an 800XL, 65XE, or 130XE, you may connect two or more printers at the same time and direct output to one or the other. From BASIC, for example, it's as simple as typing

```
LIST "P2:"
```

or

```
LIST "P5:"
```

Two major drawbacks: all printers still respond as P1:, so using P1: or just P: when two printers are attached leads to humorous and/or disastrous results. Since many programs always address printers as P:, this trick may be useful only in your own programs. Also, only the following printers have these secondary numbers:

Printer	Secondary Number
850	P2:
1025	P3:
1020	P4:
1027	P5:
1029	P6:

(The 850 refers to *any* printer attached via an 850 Interface Module. The 1029 printer is rare in the U.S.)

The fact that the 850 can handle different printer numbers indicates that provision for this feature was included as far back as 1979 (when the 850 was first made). Do you wonder why nothing was said sooner? Why don't the 400, 800, and 1200XL work with multiple printers? Do any other interface modules (from third-party vendors) have secondary device numbers? A prize for the best answer.

The Nine-Minute Nap

If you have a 1027 printer which is not lucky enough to be hooked up to an XE computer, you've probably experienced the infamous sleeping printer bug. Sometimes the 1027 just suddenly stops printing. Many people believe they need to reboot their system to wake the printer up. Actually, after about nine minutes, the printer just as suddenly springs to life again. The reasons for this are too strange and lengthy to go into here. Suffice it to say that the problem has existed since the first Atari computer was built and is related to the (also infamous) sleeping disk drive phenomenon—though the drive only sleeps about five seconds. You'll be pleased to know that Atari's newest operating system ROMs in the XE computers finally fix the problem.

If you do have a 1027, but don't have an XE, and still want to fix this problem, type in, save, and run the accompanying program. It automatically seeks out the LOMEM value for your system and then creates an AUTORUN.SYS file to patch the timeout problem. The AUTORUN.SYS file will load at that LOMEM point and then move LOMEM above itself. Since it reads the current LOMEM, be sure to create the AUTORUN.SYS file on the same disk, *booted in the same fashion*, that you later want to use. This means, for example, that any special drivers (RAM disk, RS-232, and so on) must be installed before you run this BASIC program.

For a more specific example, let's say you intend to use the 850's R: driver with *AtariWriter* and the 1027. You must start by booting the 850's AUTORUN.SYS file to install the R: driver in memory, *then* run the program below.

Also, if you have true double-density drives (not "enhanced density" 1050s), boot with double-density disks inserted. This patch should work with almost any DOS, such as DOS XL, SpartaDOS, DOS 2.5, or whatever—but I wasn't able to test them all.

Two final points: If an AUTORUN.SYS file already exists on the disk when this program is run, the 1027 patch is appended to that file. Again, using the 850 as an example, this means you'll have a single file which serves two purposes: It boots the R: driver and makes the 1027 patch. Finally, line 170 of the listing is a REMark; if you delete the REM to enable this line, it reserves two pages (512 bytes) of extra memory. If you have any trouble running this patch, try deleting the REM. For instance, if your system has more than one disk drive, you might want to make this change.

Obviously, I did not develop this program by arbitrarily typing in funny numbers for my DATA statements. I started with a program written by Joe Miller (formerly of Atari), then fixed it so that it survives SYSTEM RESET, is relocatable, moves LOMEM if appropriate, and does not install itself twice. If you're interested in studying the source code for this

program, you can download it from CompuServe. Look in the Atari eight-bit SIG's DL (DownLoad) section under utilities. The filename is P1027.FIX, and it's a document (ASCII) file.

1027 Printer Timeout Fixer

```
DD 140 REM first, find where
       LOMEM is now
KG 150 LOPAGE=PEEK(744)
BM 160 IF PEEK(743)<>0 THEN
       LDPAGE=LDPAGE+1
NJ 170 REM (see text) LOPAGE
       =LOPAGE+2
DG 180 MODE=8:TRAP 200
PM 190 OPEN #3,4,0,"D:AUTORU
       N.SYS":MODE=9
FO 200 CLOSE #3
BB 210 OPEN #3,MODE,0,"D2:AU
       TOTEST"
FE 220 IF MODE=8 THEN PUT #3
       ,255:PUT #3,255
DO 230 READ BYTE:IF BYTE<-1
       THEN 300
EM 240 IF BYTE=-1 THEN BYTE=
       LDPAGE
EO 250 PUT #3,BYTE:GOTO 230
BK 290 REM (all data in file
       )
FP 300 CLOSE #3
BL 310 END
JD 890 DATA 0,-1,80,-1
CC 900 DATA 165,49,208,19,10
       4,133,49,140
MO 910 DATA 81,-1,160,1,24,1
       77,50,101
FO 920 DATA 49,105,0,72,172,
       81,-1,76
DJ 930 DATA 80,-1,8,120,162,
       0,160,-1
BB 940 DATA 236,12,2,208,5,2
       04,13,2
MF 950 DATA 240,18,173,12,2,
       141,24,-1
MO 960 DATA 173,13,2,141,25,
       -1,142,12
NK 970 DATA 2,140,13,2,40,32
       ,80,-1
KH 980 DATA 160,-1,204,232,2
       ,144,9,200
HN 990 DATA 140,232,2,169,0,
       141,231,2
EK 1000 DATA 96
NA 1005 DATA 0,64,35,64
AL 1010 DATA 169,-1,205,232,
       2,144,28,208
MO 1020 DATA 5,173,231,2,208
       ,21,32,26
ML 1030 DATA -1,165,12,141,6
       2,-1,165,13
BC 1040 DATA 141,63,-1,169,2
       6,133,12,169
CM 1050 DATA -1,133,13,96
PA 1060 DATA 226,2,227,2,0,6
       4
PD 1070 DATA -9999
```


News & Products

New Electronic Arts Software

Electronic Arts has announced distribution of three new personal computer software packages.

Mind Mirror is a mental awareness program designed by Dr. Timothy Leary that lets you test your stereotypes by responding to various situations through the eyes of your chosen subject. You can rate any subject you like and react to various situations based on your preconceptions about the subject. This philosopher-on-a-disk is designed to let you learn about other people as well as yourself. *Mind Mirror* is available for the IBM PC, PCjr, and compatibles, the Apple II, and the Commodore 64. Suggested retail price for the IBM version is $34.95. Prices for the other versions will be available by the time you read this.

A new conquer-the-world strategy game, *Lords of Conquest*, is based on the board game *Risk*. The object of this game is to protect your holdings while trying to conquer territories belonging to your opponents. There are four levels of game complexity and an unlimited variety of game maps. *Lords of Conquest* is available for the Commodore 64 and 128 and Atari XL series for $32.95 each.

Super BoulderDash consists of the original popular arcade-style game, *BoulderDash*, and its sequel *BoulderDash II*. Both are strategic action games in which you must maneuver the hero, Rockford, through a series of caves to collect diamonds while avoiding fireflies, butterflies, and falling boulders. *BoulderDash II* adds sixteen new caves, each with five play levels. Versions are available for the Atari 400/800 and Commodore 64/128 at a suggested retail price of $22.95, and $29.95 for the Apple II and IBM PC and PCjr versions.

Electronic Arts, 1820 Gateway Dr., San Mateo, CA 94404.

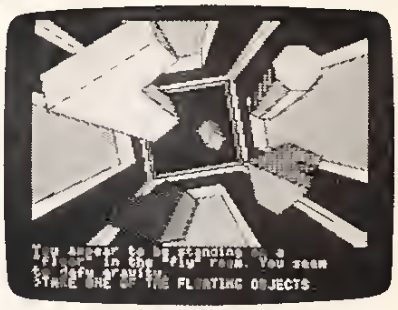
Circle Reader Service Number 200.

Commodore 64 Power Supply

The Commodore 64 Power Plus from Computer Specialties is a single AC-switched power supply with built-in surge protection. It has one on/off control and three grounded outlets to control your disk drive, monitor, and printer. It offers protection for DC power short circuits, over-current, over-temperature, surge, AC fuse, and over-voltage situations. Suggested retail price is $59.95.

Computer Specialties, Inc., P.O. Box 1718, Melbourne, FL 32902-1718.

Circle Reader Service Number 201.



Oo-Topos *from Penguin Software, a new graphics-and-text adventure game.*

Space Adventures

Your mission is to intercept a power transfusion waste spill before it destroys the earth. However, first you must escape your captors and the planetoid, Oo-Topos. That's the plot of the new adventure game *Oo-Topos* from Polarware. It's available for the Apple II and Commodore 64/128 computers at a suggested retail price of $34.95, and on the Atari ST, Macintosh, Amiga, and IBM PC and compatibles for $39.95.

Polarware, Penguin Software, 2600 Keslinger Rd., P.O. Box 311, Geneva, IL 60134.

Circle Reader Service Number 202.

Commodore Machine Language Programming

The Machine Shop is a machine language development system for the Commodore 64/128, an upgraded version of French Silk's *Develop-64* (version 4.6). It includes an integrated macro assembler, a full-screen editor, a symbolic decoder, and a debugger. All are in memory simultaneously. The system is reportedly three times faster than the popular *PAL* assembler, according to the manufacturer.

The Machine Shop from FS! Software costs $39.95, which includes an instruction manual and a free subscription to *Machine Code*, a machine language programmer's journal.

FS! Software, P.O. Box 635, Faribault, MN 55021.

Circle Reader Service Number 203.

Win A Trip To Australia

Mindscape is offering a trip to Australia for the 1987 America's Cup race as the grand prize in *The American Challenge: A Sailing Simulation* software competition. Eight finalists will each win a modem, and then compete against each other to win a trip to Perth, Australia.

The American Challenge: A Sailing Simulation is a new skill game from Mindscape that puts you onboard a racing sailboat to sail seven increasingly challenging courses until you reach the eighth course, the America's Cup race. You control sail, rudder, and centerboard while the instrument panel monitors wind speed, wind direction, and boat heading. For rookie sailors, a recorded sailing tutorial is included.

The American Challenge: A Sailing Simulation is availble for Apple II and IBM PC computers at a suggested retail price of $39.95.

Mindscape, Inc., 3444 Dundee Rd., Northbrook, IL 60062.

Circle Reader Service Number 204.

Amiga Users Group

The North American Amiga Users Group (NAAUG) is a national user organization for Amiga owners. Membership includes a subscription to the newsletter *AmigaHelp*, a helpline for free one-to-one computer advice, one free disk of public domain software, and full access to the group's public domain library, participation in the *NAAUG* co-op, and free classified ads to other members. The organization is also working toward an online Special Interest Group (SIG).

The annual membership fee is $25.

North American Amiga Users Group, Box 376, Lemont, PA 16851.

Circle Reader Service Number 205.

IBM Computer Golf

Mean 18 is a one-to-four player golf simulator game for the IBM PC/XT/AT or PCjr with 256K memory, DOS 2.1 or higher, and a color graphics adapter. It includes four graphically detailed courses, 72 different holes, a golf course architect set, and a variety of strategy and play options. The player can choose among practice tee, practice green, practice hole, and begin game options to perfect his game-playing abilities. A joystick controller is not required, although a mouse or joystick may be used.

Suggested retail price is $49.95.

Accolade, 20863 Stevens Creek Blvd., Cupertino, CA 95014.

Circle Reader Service Number 206.

New *Mastertype* Programs

Scarborough Systems has introduced Amiga and enhanced IBM versions of *Mastertype*, educational software that teaches users typing and keyboard skills. The Amiga version includes lessons on numbers, symbols, the numeric keypad, a skill test, rhythm instruction, sentence typing practice, finger positioning charts, and the *Mastertype* game. The IBM version has been upgraded to include all the features of the new Amiga version.

The Amiga and IBM versions are available for $39.95 each.

Scarborough Systems, 55 S. Broadway, Tarrytown, New York 10591.

Circle Reader Service Number 207.

Atari ST Drawing Progam

Easy-Draw from Migraph is an object-oriented drawing program for the Atari ST with a monochrome or color monitor system and the operating system in ROM. It can be used to create business graphics, presentation materials, line drawings, multiple-layer illustrations and to move objects. The program uses the GEM interface, includes standard GEM onscreen proportional text fonts, and uses high-resolution output for printing.

Suggested retial price is $149.95.

Migraph, Inc., 720 S. 333rd St., Suite 201, Federal Way, WA 98003.

Circle Reader Service Number 208.

64/128 Musical Sight Reading

MasterSoft has released *Singing Master*, a program in the company's Mastery in Music series for the Commodore 64 and 128 computers. *Singing Master* helps you learn to sight read through individualized exercises in pitch and interval awareness, notes, rhythm, and basic music facts. The program includes scales, thirds, and intervals in every major key, as well as chord analysis. There is a printer option as well.

Suggested retail price is $49.95.

MasterSoft, P.O. Box 1027, Bend, OR 97709.

Circle Reader Service Number 209.

Mystery And *Intrigue!*

Kinemation has announced a new computer mystery game, *Intrigue!*, for the Apple II and Commodore 64 computers. The action takes place in Washington, D.C., where you talk with an assortment of characters to determine who is telling the truth and who can be trusted as well as who is guilty. There are more than 2000 possible solutions and three experience levels for ages 12 to adult.

Retail price is $39.95.

Kinemation, Four Winds Rd., P.O. Box 3076, Peterborough, NH 03458-3076.

Circle Reader Service Number 210.

Scrabble En Francais

Gessler Educational Software has developed a new software program which helps you learn French while having fun. *French Micro Scrabble* is based on the board game *Scrabble* and contains a built-in vocabulary of 20,000 French words. Up to four people can play, or you can play against the computer. There are four different skill levels.

French Micro Scrabble is available for the Commodore 64/128 and Apple II+, IIe, and IIc computers for $39.95.

Gessler Educational Software, 900 Broadway, New York, NY 10003.

Circle Reader Service Number 211.

Utility Program For The Commodore 64

Disk Assistant from Spectrum 1 Network is a Commodore 64 utility program that simplifies disk commands with fifteen menu-driven disk options. Among those included are disk format, validate, erase, and rename. Other features include a help file, flexibility in accessing dual drives, copying on single as well as dual drives, and sequential data file copying.

Disk Assistant sells for $11.95.

Spectrum 1 Network, 9161 Beachy Ave., Arleta, CA 91331.

Circle Reader Service Number 212.

More Games From Mindscape

Mindscape has released three more software packages for the Commodore 64 and 128. In *Infiltrator*, your mission

is to fly through hostile enemy airspace and reach strategic targets designed to destroy the Mad Leader's military force. This adventure game combines helicopter flight simulation and military ground action.

In order to win *Spell of Destruction*, you must enter the Castle of Illusions, find the Prime Elemental, and destroy it with a single spell. This game features over 70 locations with scrolling 3-D graphics and music.

Three separate games—*Brian Bloodaxe, Revelation,* and *QuoVadis*—are combined on one disk. With *Brian Bloodaxe* you can invade Britain and seek the crown jewels. In *Revelation* you battle the Monster of the Apocalypse, or you can fight the Dark Lord in *QuoVadis*. All three are combination strategy-arcade games.

The suggested retail price for *Infiltrator* and *Spell of Destruction* is $29.95 each. The price for the three-game disk is $14.95.

Mindscape, Inc., 3444 Dundee Rd., Northbrook, IL 60062.

Circle Reader Service Number 213.

ST Mind Game

Brøderbund has released an Atari ST version of Synapse's *Mindwheel*, a science fiction text adventure game that features a vocabulary of 1,200 to 1,500 words. The game is a time journey through the minds of four people—a peace activist rock star, a monstrous dictator, a heroic poet, and a gifted scientist. The action takes place in real time, and the goal is to retrieve the Wheel of Wisdom.

Mindwheel is available for the Atari 520ST for $44.95. Versions are also available for the IBM PC/PCjr, Apple II, and Macintosh computers for $44.95 each, and for the Commodore 64/128 and Atari 400/800/XL/XE computers for $39.95.

Brøderbund Software, 17 Paul Dr., San Rafael, CA 94903-2101.

Circle Reader Service Number 214.

Educational Software From MECC

Minnesota Educational Computing Corporation (MECC) has announced two new learning programs. In *Number Munchers*, students move a number muncher around to devour number expressions that match a value displayed on the screen, while avoiding the predatory "troggles". There are five different versions of this game on the disk. For grades four to eight.

MECC Dataquest: The Fifty States teaches students to form questions about the fifty states, look for answers in a database with a menu-driven search program, and formulate hypotheses by using the search results. For grades five to eleven.

Both packages are designed for Apple II computers with at least 64K memory, and are priced at $49 each.

Minnesota Educational Computing Corporation, 3490 Lexington Ave. N., Saint Paul, MN 55126-8097.

Circle Reader Service Number 215.

Computer Baseball

With *Monday Morning Manager, The Baseball Game* you can play any major league baseball team against any other team. The 1986 revised version includes 64 major league teams from 1905 through the 1985 playoff teams with over 1,500 players and pitchers. The results of each play are based upon the actual statistics of the players, and each play is graphically displayed on your screen.

Monday Morning Manager is available on the Atari 800 and Commodore 64 for $39.95, on the Apple II for $44.95, and on the Atari 520ST and IBM PC for $50.

TK Computer Products, P.O. Box 9617, Downers Grove, IL 60515; distributed by Computer Software Service, 495A Busse Rd., Elk Grove Village, IL 60007.

Circle Reader Service Number 216.

Atari ST Backgammon

Hippopotamus Software has introduced *HippoBackgammon*, a programmable backgammon game which teaches artificial intelligence theory. *HippoBackgammon* allows users to modify the artificial intelligence of the game's built-in opponents to test strategies and playing styles. The strategy is based on statistics which predict the probability of winning with certain moves. There are three levels for novice, intermediate, or expert play. The game works on the Atari ST with either color or black and white monitor.

Retail price is $39.95.

Hippopotamus Software, Inc., 985 University Ave., Suite #12, Los Gatos, CA 95030.

Circle Reader Service Number 217.

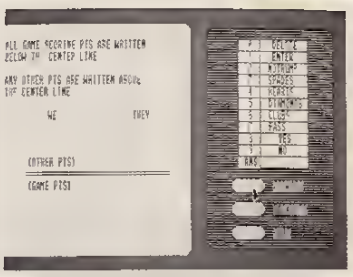
Graphics Strategy Game

Infocom has released *Fooblitzky*, a multiplayer, computer graphics strategy game. The object of the game is to find the four secret items in the city of Fooblitzky. The challenge is that you are a *cool canine* and the four secret objects change every time you play the game. Each package contains a disk, four workboards, four markers, and two sets of rules.

Fooblitzky is available for the Atari XL/XE computers with 48K RAM and 810 or 1050 disk drive, the Apple II series with 128K, or the IBM PC with 128K and graphics card. Each version retails for $39.95, and can be used with joystick or keyboard.

Infocom, Inc., 125 CambridgePark Dr., Cambridge, MA 02140.

Circle Reader Service Number 218.



Compubridge, *a bridge tutorial for the Atari 520ST and 1040ST from Artworx.*

Bridge Tutorial For ST Computers

Artworx has begun shipping *Compubridge* for the Atari 520 and 1040 ST.

Based on the bridge text, *Contract Bridge, Five-Card Major Approach*, by Silverman, Jais and Lebel, the program consists of ten chapters covering all aspects of the game from the basics to the game's more sophisticated points. Eight of the chapters close with brief quizzes that test your knowledge. Each quiz is randomly generated, which may be especially helpful for the advanced player interested in fine-tuning his or her game. All user input is through the ST's mouse; no keyboard entry is required.

Suggested retail price of *Compubridge* is $29.95.

Artworx Software Company, Inc., 150 N. Main St., Fairport, NY 14450.

Circle Reader Service Number 219.

Commodore, Apple II Educational Software

Balance! is an interactive program that teaches students about solving equations. By working with graphically displayed linear equations on a "balance beam," students can literally see the equation-solving process, and better understand it. It's aimed at beginning algebra students, advanced students who want to review the basics, and parents and teachers looking for more effective ways of teaching basic math concepts.

Developed at the New York Institute of Technology, the program strives

to teach what an equation is and how it works; understand that a solution is a unique value which, when inserted in the equation, causes its two sides to be equal; and to develop a strategy, or algorithm, for solving equations and to understand how and why the algorithm works.

Available for Apple II and Commodore 64, *Balance!* retails for $49.

HRM Software, 175 Tompkins Ave., Pleasantville, NY 10570.

Circle Reader Service Number 220.

SpeedScript Enhancer For 64

Upstart Publishing has released *SpeedMate*, a customization program for COMPUTE! Publications' *SpeedScript 3.0–3.2* for the Commodore 64. *SpeedMate* lets you control the way text appears on the screen while editing. It also customizes the control commands for *Preview 80*, an 80-column *SpeedScript* page preview program. (*SpeedScript* and *Preview 80* are not included with *SpeedMate*, but are available from COMPUTE! Publications.) *SpeedMate* also includes an optional print preview routine which displays 80 columns of text with no horizontal scrolling.

SpeedMate is available for $15.

Upstart Publishing, Dept. NPMC, P.O. Box 22022, Greensboro, NC 27420.

Circle Reader Service Number 221.

IBM Graphics Software For Daisywheel Printers

Daisyfont, from Einstein's Automation Profiles is a program that provides dot-matrix design and print capabilities for all daisywheel printers and spinwriters with IBM PC computers. No hardware modifications are necessary.

Daisyfont resides in memory, and can be called up from within any other application software. The program lets you design and print logos, letterheads, report headlines, borders, special fonts, and custom character sets. Suggested retail price is $69.95.

Einstein's Automation Profiles, Inc., 184 2nd Ave., #1B, New York, NY 10003.

Circle Reader Service Number 222.

Commodore, Apple II Integrated Software

Software Resource Group has begun shipping *Brown Bag Software*, an inexpensive integrated word processor/database manager for the Commodore 64 and Apple II (both versions are on flip sides of the same disk). This program lets you incorporate information from your databases into letters and reports easily and quickly. Up to 20 of these merges can be done within any one document.

Editing features of the word processor include global search and replace, headers, trailers, footers, and delete by character, word, line, and paragraph. The database manager allows you to create your own templates or use the ones provided.

Brown Bag Software retails for $59.95.

Software Resource Group, Inc., 15100 El Camino Grande, Saratoga, CA 95070.

Circle Reader Service Number 223.

Accelerating The Apple

A new high-speed replacement coprocessor from Titan Technologies can triple the speed of your Apple II, IIe, or II+. Called the Accelerator IIe, it has its own 6502 processor and plugs into any slot. It can be used to increase the speed of *AppleWorks, Apple Writer, FlashCalc, Multiplan*, and other business applications. If you need to run a program at normal speed, you can slow the Accelerator IIe down with your preboot disk. Suggested retail price, $319.

Titan Technologies, Inc., 310 West Ann St., Ann Arbor, MI 48104-1337.

Circle Reader Service Number 224.

The HabaDisk ten-megabyte hard disk drive for the Atari ST sells for $699.95

Haba Hard Drive For ST

Haba/Arrays has announced an external ten-megabyte hard disk drive for the Atari ST, priced at $699.95. The HabaDisk is a plug-in disk and stores the equivalent of more than twelve dual-sided 800K disks. Transfer rate is five megabytes a second.

The drive is self-powered, and an Atari interface cable is included.

Haba/Arrays, Inc., 6711 Valjean Ave., Van Nuys, CA 91406.

Circle Reader Service Number 225.

Mac Digitized Images On A Disk

RealArt, from Electronic Cottage Industries, is a disk for the Macintosh that contains just under 400K of digitized artwork. You can preview the artwork by running the public domain slide show, and then look more closely at your favorites. Then shrink, move, cut, or copy and print out the desired images for use in letterhead stationery, note paper, drawing education, or framed display.

RealArt retails for $29.95.

Electronic Cottage Industries, P.O. Box 217, Spooner, WI 54801.

Circle Reader Service Number 226.

Vietnam Strategy Game

MicroProse Software has introduced *Conflict In Vietnam*, a strategic simulation of the crucial battles of the Vietnam War, available for the Commodore 64 and 128, Atari XL/XE series, Apple II family, and IBM PCjr computers.

Five separate games are included in the program, with scenarios ranging from the end of French rule at Dien Bien Phu in 1954 to the North Vietnamese assault on Quang Tri in 1972. The three battles in between illustrate various stages of American involvement: Ia Drang (1965), Khe Sanh (1968), and Cambodia (1970). The scenarios can be played independently or in historical order. Commands can be entered by joystick or from the keyboard.

A 110-page manual is included, with quick-start instructions, detailed information for advanced play, extensive historical background, design notes, play tips, maps, and charts. Two people can play each other, or one person can play against the computer. There is an option to take command of the North Vietnamese side and play against the computer-controlled American forces.

The suggested retail price is $39.95 for each version.

Microprose Software, Inc., 120 Lakefront Dr., Hunt Valley, MD 21030.

Circle Reader Service Number 227.

ST Database

Mirage Concepts has introduced *H & D Base*, a relational database management language for the Atari ST computers. The program is a *dBase II* work-alike with almost 300 commands available for the manipulation of data. In addition to regular data storage and retrieval, *H & D Base* can be used for the creation of systems for handling inventories, accounts payable and receivable, client lists, and more.

The suggested retail price is $99.95, and the program is not copy protected.

Mirage Concepts, Inc., 4055 W. Shaw #108, Fresno, CA 93711.

Circle Reader Service Number 228.

©

COMPUTE!'s Guide To Typing In Programs

Computers are precise—type the program *exactly* as listed, including necessary punctuation and symbols, except for special characters noted below. We have provided a special listing convention as well as a program to check your typing—"The Automatic Proofreader."

Programs for the IBM, TI-99/4A, and Atari ST models should be typed exactly as listed; no special characters are used. Programs for Commodore, Apple, and Atari 400/800/XL/XE computers may contain some hard-to-read special characters, so we have a listing system that indicates these control characters. You will find these Commodore and Atari characters in curly braces; *do not type the braces.* For example, {CLEAR} or {CLR} instructs you to insert the symbol which clears the screen on the Atari or Commodore machines. A complete list of these symbols is shown in the tables below. For Commodore, Apple, and Atari, a single symbol by itself within curly braces is usually a control key or graphics key. If you see {A}, hold down the CONTROL key and press A. This will produce a reverse video character on the Commodore (in quote mode), a graphics character on the Atari, and an invisible control character on the Apple.

Graphics characters entered with the Commodore logo key are enclosed in a special bracket: [<A>]. In this case, you would hold down the Commodore logo key as you type A. Our Commodore listings are in uppercase, so shifted symbols are underlined. A graphics heart symbol (SHIFT-S) would be listed as S. One exception is {SHIFT-SPACE}. When you see this, hold down SHIFT and press the space bar. If a number precedes a symbol, such as {5 RIGHT}, {6 S}, or [<8 Q>], you would enter five cursor rights, six shifted S's, or eight Commodore-Q's. On the Atari, inverse characters (white on black) should be entered with the inverse video

Atari 400/800/XL/XE

When you see	Type	See	
{CLEAR}	ESC SHIFT <		Clear Screen
{UP}	ESC CTRL -		Cursor Up
{DOWN}	ESC CTRL =		Cursor Down
{LEFT}	ESC CTRL +		Cursor Left
{RIGHT}	ESC CTRL *		Cursor Right
{BACK S}	ESC DELETE		Backspace
{DELETE}	ESC CTRL DELETE		Delete character
{INSERT}	ESC CTRL INSERT		Insert character
{DEL LINE}	ESC SHIFT DELETE		Delete line
{INS LINE}	ESC SHIFT INSERT		Insert line
{TAB}	ESC TAB		TAB key
{CLR TAB}	ESC CTRL TAB		Clear tab
{SET TAB}	ESC SHIFT TAB		Set tab stop
{BELL}	ESC CTRL 2		Ring buzzer
{ESC}	ESC ESC		ESCape key

Commodore PET/CBM/VIC/64/128/16/+4

When You Read:	Press:	See:
{CLR}	SHIFT + CLR/HOME	
{HOME}	CLR/HOME	
{UP}	SHIFT + ↑ CRSR ↓	
{DOWN}	↑ CRSR ↓	
{LEFT}	SHIFT + ← CRSR →	
{RIGHT}	← CRSR →	
{RVS}	CTRL 9	
{OFF}	CTRL 0	
{BLK}	CTRL 1	
{WHT}	CTRL 2	
{RED}	CTRL 3	
{CYN}	CTRL 4	
{PUR}	CTRL 5	
{GRN}	CTRL 6	
{BLU}	CTRL 7	
{YEL}	CTRL 8	

When You Read:	Press:	See:
{ 1 }	COMMODORE 1	
{ 2 }	COMMODORE 2	
{ 3 }	COMMODORE 3	
{ 4 }	COMMODORE 4	
{ 5 }	COMMODORE 5	
{ 6 }	COMMODORE 6	
{ 7 }	COMMODORE 7	
{ 8 }	COMMODORE 8	
{ F1 }	f1	
{ F2 }	SHIFT + f1	
{ F3 }	f3	
{ F4 }	SHIFT + f3	
{ F5 }	f5	
{ F6 }	SHIFT + f5	
{ F7 }	f7	
{ F8 }	SHIFT + f7	
←	←	

key (Atari logo key on 400/800 models).

Whenever more than two spaces appear in a row, they are listed in a special format. For example, {6 SPACES} means press the space bar six times. Our Commodore listings never leave a single space at the end of a line, instead moving it to the next printed line as {SPACE}.

Amiga program listings contain only one special character, the left arrow (←) symbol. This character marks the end of each program line. Wherever you see a left arrow, press RETURN or move the cursor off the line to enter that line into memory. Don't try to type in the left arrow symbol; it's there only as a marker to indicate where each program line ends.

The Automatic Proofreader

Type in the appropriate program listed below, then save it for future use. The Commodore Proofreader works on the Commodore 128, 64, Plus/4, 16, and VIC-20. Don't omit any lines, even if they contain unfamiliar commands or you think they don't apply to your computer. When you run the program, it installs a machine language program in memory and erases its BASIC portion automatically (so be sure to save several copies before running the program for the first time). If you're using a Commodore 128, Plus/4 or 16, do *not* use any GRAPHIC commands while the Proofreader is active. You should disable the Commodore Proofreader before running any other program. To do this, either turn the computer off and on or enter SYS 64738 (for the 64), SYS 65341 (128), SYS 64802 (VIC-20), or SYS 65526 (Plus/4 or 16). To reenable the Proofreader, reload the program and run it as usual. Unlike the original VIC/64 Proofreader, this version works the same with disk or tape.

On the Atari, run the Proofreader to activate it (the Proofreader remains active in memory as a machine language program); you must then enter NEW to erase the BASIC loader. Pressing SYSTEM RESET deactivates the Atari Proofreader; enter PRINT USR(1536) to reenable it.

The Apple Proofreader erases the BASIC portion of itself after you run it, leaving only the machine language portion in memory. It works with either DOS 3.3 or ProDOS. Disable the Apple Proofreader by pressing CTRL-RESET before running another BASIC program.

The IBM Proofreader is a BASIC program that simulates the IBM BASIC line editor, letting you enter, edit, list, save, and load programs that you type. Type RUN to activate. Be sure to leave Caps Lock on, except when typing lowercase characters.

Once the Proofreader is active, try typing in a line. As soon as you press RETURN, either a hexadecimal number (on the Apple) or a pair of letters (on the Commodore, Atari, or IBM) appears. The number or pair of letters is called a *checksum*.

Compare the value displayed on the screen by the Proofreader with the checksum printed in the program listing in the magazine. The checksum is given to the left of each line number. Just type in the program a line at a time (without the printed checksum), press RETURN or Enter, and compare the checksums. If they match, go on to the next line. If not, check your typing; you've made a mistake. Because of the checksum method used, do not type abbreviations, such as ? for PRINT. On the Atari and Apple Proofreaders, spaces are not counted as part of the checksum, so be sure you type the right number of spaces between quote marks. The Atari Proofreader does not check to see that you've typed the characters in the right order, so if characters are transposed, the checksum still matches the listing. The Commodore Proofreader catches transposition errors and ignores spaces unless they're enclosed in quotation marks. The IBM Proofreader detects errors in spacing and transposition.

IBM Proofreader Commands

Since the IBM Proofreader replaces the computer's normal BASIC line editor, it has to include many of the direct-mode IBM BASIC commands. The syntax is identical to IBM BASIC. Commands simulated are LIST, LLIST, NEW, FILES, SAVE, and LOAD. When listing your program, press any key (except Ctrl-Break) to stop the listing. If you enter NEW, the Proofreader prompts you to press Y to be especially sure you mean yes.

Two new commands are BASIC and CHECK. BASIC exits the Proofreader back to IBM BASIC, leaving the Proofreader in memory. CHECK works just like LIST, but shows the checksums along with the listing. After you have typed in a program, save it to disk. Then exit the Proofreader with the BASIC command, and load the program as usual (this replaces the Proofreader in memory). You can now run the program, but you may want to resave it to disk. This will shorten it on disk and make it load faster, but it can no longer be edited with the Proofreader. If you want to convert an existing BASIC program to Proofreader format, save it to disk with SAVE *"filename"*,A.

Program 1: Atari Proofreader

By Charles Brannon, Program Editor

```
100 GRAPHICS 0
110 FOR I=1536 TO 1700:REA
    D A:POKE I,A:CK=CK+A:N
    EXT I
120 IF CK<>19072 THEN ? "E
    rror in DATA Statement
    s.  Check Typing.":END

130 A=USR(1536)
140 ? :? "Automatic Proofr
    eader Now Activated."
150 END
160 DATA 104,160,0,185,26,
    3,201,69,240,7
170 DATA 200,200,192,34,20
    8,243,96,200,169,74
180 DATA 153,26,3,200,169,
    6,153,26,3,162
190 DATA 0,189,0,228,157,7
    4,6,232,224,16
200 DATA 208,245,169,93,14
    1,78,6,169,6,141
210 DATA 79,6,24,173,4,228
    ,105,1,141,95
220 DATA 6,173,5,228,105,0
    ,141,96,6,169
230 DATA 0,133,203,96,247,
    238,125,241,93,6
240 DATA 244,241,115,241,1
    24,241,76,205,238
250 DATA 0,0,0,0,0,32,62,2
    46,8,201
260 DATA 155,240,13,201,32
    ,240,7,72,24,101
270 DATA 203,133,203,104,4
    0,96,72,152,72,138
280 DATA 72,160,0,169,128,
    145,88,200,192,40
290 DATA 208,249,165,203,7
    4,74,74,74,24,105
300 DATA 161,160,3,145,88,
    165,203,41,15,24
310 DATA 105,161,200,145,8
    8,169,0,133,203,104
320 DATA 170,104,168,104,4
    0,96
```

Program 2: IBM Proofreader

By Charles Brannon, Program Editor

```
10 'Automatic Proofreader Vers
   ion 3.0 (Lines 205,206 adde
   d/190 deleted/470,490 chang
   ed from V2.0)
100 DIM L$(500),LNUM(500):COLO
    R 0,7,7:KEY OFF:CLS:MAX=0:
    LNUM(0)=65536!
110 ON ERROR GOTO 120:KEY 15,C
    HR$(4)+CHR$(70):ON KEY(15)
     GOSUB 640:KEY (15) ON:GOT
    O 130
120 RESUME 130
130 DEF SEG=&H40:W=PEEK(&H4A)
140 ON ERROR GOTO 650:PRINT:PR
    INT"Proofreader Ready."
150 LINE INPUT L$:Y=CSRLIN-INT
    (LEN(L$)/W)-1:LOCATE Y,1
160 DEF SEG=0:POKE 1050,30:POK
    E 1052,34:POKE 1054,0:POKE
     1055,79:POKE 1056,13:POKE
     1057,28:LINE INPUT L$:DEF
    SEG:IF L$="" THEN 150
170 IF LEFT$(L$,1)=" " THEN L$
    =MID$(L$,2):GOTO 170
```

```
1B0 IF VAL(LEFT$(L$,2))=0 AND
    MID$(L$,3,1)=" " THEN L$=M
    ID$(L$,4)
200 IF ASC(L$)>57 THEN 260 'no
     line number, therefore co
    mmand
205 BL=INSTR(L$," "):IF BL=0 T
    HEN BL$=L$:GDTO 206 ELSE B
    L$=LEFT$(L$,BL-1)
206 LNUM=VAL(BL$):TEXT$=MID$(L
    $,LEN(STR$(LNUM))+1)
210 IF TEXT$="" THEN GOSUB 540
    :IF LNUM=LNUM(P) THEN GDSU
    B 560:GOTO 150 ELSE 150
220 CKSUM=0:FOR I=1 TO LEN(L$)
    :CKSUM=(CKSUM+ASC(MID$(L$,
    I))*I) AND 255:NEXT:LDCATE
     Y,1:PRINT CHR$(65+CKSUM/1
    6)+CHR$(65+(CKSUM AND 15))
    +" "+L$
230 GOSUB 540:IF LNUM(P)=LNUM
    THEN L$(P)=TEXT$:GDTO 150
    'replace line
240 GDSUB 580:GDTD 150 'insert
     the line
260 TEXT$="":FOR I=1 TO LEN(L$
    ):A=ASC(MIO$(L$,I)):TEXT$=
    TEXT$+CHR$(A+32*(A>96 AND
    A<123)):NEXT
270 OELIMITER=INSTR(TEXT$," ")
    :CDMMAND$=TEXT$:ARG$="":IF
     DELIMITER THEN COMMAND$=L
    EFT$(TEXT$,OELIMITER-1):AR
    G$=MID$(TEXT$,DELIMITER+1)
     ELSE OELIMITER=INSTR(TEXT
    $,CHR$(34)):IF DELIMITER T
    HEN CDMMAND$=LEFT$(TEXT$,D
    ELIMITER-1):ARG$=MIO$(TEXT
    $,DELIMITER)
2B0 IF COMMANO$<>"LIST" THEN 4
    10
290 OPEN "scrn:" FOR DUTPUT AS
     #1
300 IF ARG$="" THEN FIRST=0:P=
    MAX-1:GOTD 340
310 OELIMITER=INSTR(ARG$,"-"):
    IF DELIMITER=0 THEN LNUM=V
    AL(ARG$):GOSUB 540:FIRST=P
    :GDTD 340
320 FIRST=VAL(LEFT$(ARG$,OELIM
    ITER)):LAST=VAL(MIO$(ARG$,
    DELIMITER+1))
330 LNUM=FIRST:GOSUB 540:FIRST
    =P:LNUM=LAST:GOSUB 540:IF
    P=0 THEN P=MAX-1
340 FDR X=FIRST TO P:N$=MIO$(S
    TR$(LNUM(X)),2)+" "
350 IF CKFLAG=0 THEN A$="":GOT
    D 370
360 CKSUM=0:A$=N$+L$(X):FOR I=
    1 TO LEN(A$):CKSUM=(CKSUM+
    ASC(MIO$(A$,I))*I) AND 255
    :NEXT:A$=CHR$(65+CKSUM/16)
    +CHR$(65+(CKSUM AND 15))+"
     "
370 PRINT #1,A$+N$+L$(X)
3B0 IF INKEY$<>"" THEN X=P
390 NEXT :CLOSE #1:CKFLAG=0
400 GOTD 130
410 IF COMMAND$="LLIST" THEN O
    PEN "Ipt1:" FOR DUTPUT AS
    #1:GOTO 300
420 IF COMMANO$="CHECK" THEN C
    KFLAG=1:GDTD 290
430 IF CDMMANO$<>"SAVE" THEN 4
    50
440 GOSUB 600:OPEN ARG$ FOR OU
    TPUT AS #1:ARG$="":GOTO 30
    0
450 IF COMMAND$<>"LOAO" THEN 4
    90
460 GDSUB 600:DPEN ARG$ FOR IN
    PUT AS #1:MAX=0:P=0
470 WHILE NOT EOF(1):LINE INPU
    T #1,L$:BL=INSTR(L$," "):B
    L$=LEFT$(L$,BL-1):LNUM(P)=
    VAL(BL$):L$(P)=MID$(L$,LEN
    (STR$(VAL(BL$)))+1):P=P+1:
    WEND
4B0 MAX=P:CLDSE #1:GDTD 130
490 IF CDMMAND$="NEW" THEN INP
    UT "Erase program - Are yo
    u sure";L$:IF LEFT$(L$,1)=
    "y" DR LEFT$(L$,1)="Y" THE
    N MAX=0:LNUM(0)=65536!:GDT
    D 130:ELSE 130
500 IF COMMANO$="BASIC" THEN C
    DLOR 7,0,0:DN ERRDR GDTD 0
    :CLS:ENO
510 IF COMMAND$<>"FILES" THEN
    520
515 IF ARG$="" THEN ARG$="A:"
    ELSE SEL=1:GDSUB 600
517 FILES ARG$:GOTO 130
520 PRINT"Syntax error":GDTD 1
    30
540 P=0:WHILE LNUM>LNUM(P) ANO
     P<MAX:P=P+1:WEND:RETURN
560 MAX=MAX-1:FOR X=P TO MAX:L
    NUM(X)=LNUM(X+1):L$(X)=L$(
    X+1):NEXT:RETURN
580 MAX=MAX+1:FOR X=MAX TO P+1
     STEP -1:LNUM(X)=LNUM(X-1)
    :L$(X)=L$(X-1):NEXT:L$(P)=
    TEXT$:LNUM(P)=LNUM:RETURN
600 IF LEFT$(ARG$,1)<>CHR$(34)
     THEN 520 ELSE ARG$=MID$(A
    RG$,2)
610 IF RIGHT$(ARG$,1)=CHR$(34)
     THEN ARG$=LEFT$(ARG$,LEN(
    ARG$)-1)
620 IF SEL=0 ANO INSTR(ARG$,".
    ")=0 THEN ARG$=ARG$+".BAS"
630 SEL=0:RETURN
640 CLOSE #1:CKFLAG=0:PRINT"St
    opped.":RETURN 150
650 PRINT "Error #";ERR:RESUME
     150
```

Program 3: Commodore Proofreader

By Philip Nelson, Assistant Editor

```
10 VEC=PEEK(772)+256*PEEK(773)
   :LO=43:HI=44
20 PRINT "AUTOMATIC PROOFREADE
   R FOR ";:IF VEC=42364 THEN
   {SPACE}PRINT "C-64"
30 IF VEC=50556 THEN PRINT "VI
   C-20"
40 IF VEC=35158 THEN GRAPHIC C
   LR:PRINT "PLUS/4 & 16"
50 IF VEC=17165 THEN LO=45:HI=
   46:GRAPHIC CLR:PRINT"128"
60 SA=(PEEK(LO)+256*PEEK(HI))+
   6:ADR=SA
70 FOR J=0 TO 166:READ BYT:POK
   E ADR,BYT:ADR=ADR+1:CHK=CHK
   +BYT:NEXT
80 IF CHK<>20570 THEN PRINT "*
    ERROR* CHECK TYPING IN DATA
    STATEMENTS":END
90 FOR J=1 TO 5:READ RF,LF,HF:
   RS=SA+RF:HB=INT(RS/256):LB=
   RS-(256*HB)
100 CHK=CHK+RF+LF+HF:POKE SA+L
   F,LB:POKE SA+HF,HB:NEXT
110 IF CHK<>22054 THEN PRINT "
   *ERROR* RELOAD PROGRAM AND
   {SPACE}CHECK FINAL LINE":EN
   D
120 POKE SA+149,PEEK(772):POKE
    SA+150,PEEK(773)
130 IF VEC=17165 THEN POKE SA+
   14,22:POKE SA+18,23:POKESA+
   29,224:POKESA+139,224
140 PRINT CHR$(147);CHR$(17);"
   PROOFREADER ACTIVE":SYS SA
150 POKE HI,PEEK(HI)+1:POKE (P
   EEK(LO)+256*PEEK(HI))-1,0:N
   EW
160 DATA 120,169,73,141,4,3,16
   9,3,141,5,3
170 DATA 88,96,165,20,133,167,
   165,21,133,168,169
180 DATA 0,141,0,255,162,31,18
   1,199,157,227,3
190 DATA 202,16,248,169,19,32,
   210,255,169,18,32
200 DATA 210,255,160,0,132,180
   ,132,176,136,230,180
210 DATA 200,185,0,2,240,46,20
   1,34,208,8,72
220 DATA 165,176,73,255,133,17
   6,104,72,201,32,208
230 DATA 7,165,176,208,3,104,2
   08,226,104,166,180
240 DATA 24,165,167,121,0,2,13
   3,167,165,168,105
250 DATA 0,133,168,202,208,239
   ,240,202,165,167,69
260 DATA 168,72,41,15,168,185,
   211,3,32,210,255
270 DATA 104,74,74,74,74,168,1
   85,211,3,32,210
280 DATA 255,162,31,189,227,3,
   149,199,202,16,248
290 DATA 169,146,32,210,255,76
   ,86,137,65,66,67
300 DATA 68,69,70,71,72,74,75,
   77,80,81,82,83,88
310 DATA 13,2,7,167,31,32,151,
   116,117,151,128,129,167,136
   ,137
```

Program 4: Apple Proofreader

By Tim Victor, Editorial Programmer

```
10 C = 0: FOR I = 76B TO 76B +
   6B: READ A:C = C + A: POKE I
   ,A: NEXT
20 IF C < > 725B THEN PRINT "ER
   ROR IN PROOFREADER DATA STAT
   EMENTS": END
30 IF PEEK (190 * 256) < > 76 T
   HEN POKE 56,0: POKE 57,3: CA
   LL 1002: GOTO 50
40 PRINT CHR$ (4);"IN#A$300"
50 POKE 34,0: HOME : POKE 34,1:
    VTAB 2: PRINT "PROOFREADER
   INSTALLED"
60 NEW
100 DATA 216,32,27,253,201,141
110 DATA 208,60,13B,72,169,0
120 DATA 72,189,255,1,201,160
130 DATA 240,8,104,10,125,255
140 DATA 1,105,0,72,202,208
150 DATA 238,104,170,41,15,9
160 DATA 48,201,58,144,2,233
170 DATA 57,141,1,4,138,74
180 DATA 74,74,74,41,15,9
190 DATA 48,201,58,144,2,233
200 DATA 57,141,0,4,104,170
210 DATA 169,141,96
```

MLX Machine Language Entry Program For Commodore 64 and Apple

Ottis Cowper, Technical Editor and Tim Victor, Editorial Programmer

"MLX" is a labor-saving utility that allows almost fail-safe entry of machine language programs. The Apple version runs on the II, II+, IIe, and IIc, with either DOS 3.3 or ProDOS.

"MLX" is a new way to enter long machine language (ML) programs without a lot of fuss. MLX lets you enter the numbers from a special list that looks similar to BASIC DATA statements. It checks your typing on a line-by-line basis. It won't let you enter invalid characters or let you continue if there's a mistake in a line. It won't even let you enter a line or digit out of sequence. For the Commodore 64, this new version of MLX was first introduced in the December 1985 issue. No version of 64 MLX published before that date can be used to enter the MLX-format listings in this issue.

Using MLX

Type in and save some copies of whichever version of MLX is appropriate for your computer (you'll want to use it to enter future ML programs from COMPUTE!). Program 1 is for the Commodore 64, and Program 2 is for the Apple. For Apple MLX, it doesn't matter whether you save the program on a disk formatted for DOS 3.3 or ProDOS. Programs entered with Apple MLX, however, must be saved to a disk formatted with the same operating system as MLX itself. If you have an Apple IIe or IIc, make sure that the key marked *Caps Lock* is in the down position.

When you're ready to enter an ML program, load and run MLX. It asks you for a starting address and an ending address. These addresses appear in the article accompanying the MLX-format program listing you're typing. If you're unfamiliar with machine language, the addresses (and all other values you enter in MLX) may appear strange. Instead of the usual decimal numbers you're accustomed to, these numbers are in *hexadecimal*—a base 16 numbering system commonly used by ML programmers. Hexadecimal—hex for short—includes the numerals 0–9 and the letters A–F. But don't worry—even if you know nothing about ML or hex, you should have no trouble using MLX.

After you enter the starting and ending addresses, the 64 version will offer you the option of clearing the workspace. Choose this option if you're starting to enter a new listing. If you're continuing a listing that's partially typed from a previous session, don't choose this option.

A functions menu will appear. The first option in the menu is ENTER DATA. If you're just starting to type in a program, pick this. Press the E key, and type the first number in the first line of the program listing. If you've already typed in part of a program, type the line number where you left off typing at the end of the previous session. In any case, make sure the address you enter corresponds to the address of a line in the listing you are entering. Otherwise, you'll be unable to enter the data correctly. In the 64 version, if you pressed E by mistake, you can return to the command menu by pressing RETURN alone when asked for the address. (You can get back to the menu from most options by pressing RETURN with no other input.)

Once you're in Enter mode, MLX prints the address for each program line for you. You then type in all nine numbers on that line, beginning with the first two-digit number after the colon (:). Each line represents eight data bytes and a checksum. Although an MLX-format listing appears similar to the "hex dump" machine language listings you may be accustomed to, the extra checksum number on the end allows MLX to check your typing. (Apple users *can* enter the data from an MLX listing using the built-in monitor if the rightmost column of data is omitted, but we recommend against it. It's much easier to let MLX do the proofreading and error checking for you.)

When you enter a line, MLX recalculates the checksum from the eight bytes and the address and compares this value to the number from the ninth column. If the values match, the data is added to the workspace area, and the prompt for the next line of data appears (the 64 version gives a pleasant beep to indicate that the line was entered correctly). But if MLX detects a typing error, you'll be notified of the mistake. The 64 version will sound a low buzz and display an error message, then redisplay the line for editing. Apple MLX sounds a beep to alert you of the error and then erases the incorrect line and prompts you to reenter it correctly.

After you have entered the last number on the last line of the listing, the Apple version will return to the command menu. At this point you should immediately choose the option S to save your data. The 64 version automatically moves to the Save option after the last number is entered.

Invalid Characters Banned

In 64 MLX, only a few keys are active while you're entering data, so you may have to unlearn some habits. You *do not* type spaces between the columns; the new MLX automatically inserts these for you. You *do not* press RETURN after typing the last number in a line; the new MLX automatically enters and checks the line after you type the last digit.

Apple MLX is fairly flexible about how you type in the numbers. You can put extra spaces between numbers or leave the spaces out entirely, compressing a line into 18 keypresses. But be careful not to put a space between two digits in the middle of a number. MLX will read two single-digit numbers instead of one two-digit number (F 6 means F *and* 6, not F6). You must press RETURN to enter the line.

Only the numerals 0–9 and the letters A–F can be typed in. If you press any other key (with some exceptions noted below), nothing happens (the 64 version gives a warning buzz to indicate an invalid keypress). Even better, MLX checks for transposed characters. If you're supposed to type in *A0* and instead enter *0A*, MLX will catch your mistake.

Editing Features

To correct typing mistakes before finishing a line in the 64 version, use the INST/DEL key to delete the character to the left of the cursor. (The cursor-left key also deletes.) If you mess up a line really badly, press CLR/HOME to start the line over. The RETURN key is also active, but only before any data is typed on a line. Pressing RETURN at this point returns you to the command menu. After you type a character of data, MLX disables RETURN until the cursor returns to the start of a line. Remember, you can press CLR/HOME to quickly get to a line number prompt.

More editing features are available when correcting lines in which 64 MLX has detected an error. To make corrections in a line that MLX has redisplayed for editing, compare the line on the

screen with the one printed in the listing, then move the cursor to the mistake and type the correct key. The cursor left and right keys provide the normal cursor controls. (The INST/DEL key now works as an alternative cursor-left key.) You cannot move left beyond the first character in the line. If you try to move beyond the rightmost character, you'll reenter the line. During editing, RETURN is active; pressing it tells MLX to recheck the line. You can press the CLR/HOME key to clear the entire line if you want to start from scratch, or if you want to get to a line number prompt to use RETURN to get back to the menu.

Apple MLX also includes some editing features. The left- and right-arrow keys allow you to back up and go forward on the line you're entering so that you can retype data. Pressing the CONTROL (CTRL) and D keys at the same time (*delete*) removes the character under the cursor, shortening the line by one character. Pressing CONTROL-I (*insert*) puts a space under the cursor and shifts the rest of the line to the right, making the line one character longer. If the cursor is at the right end of the line, neither CONTROL-D nor CONTROL-I has any effect. To leave Enter mode, press the RETURN key when MLX prompts you with a new line address.

Display Data

The second menu choice, DISPLAY DATA, examines memory and shows the contents in the same format as the program listing (including the checksum). When you press D, MLX asks you for a starting address. Be sure that the starting address you give corresponds to a line number in the listing. Otherwise, the checksum display will be meaningless. MLX displays program lines until it reaches the end of the program, at which point the menu is redisplayed. With Apple MLX, you can stop the display and return to the menu by pressing any key. The 64 version allows you to stop the display and get back to the menu by pressing RETURN, or to pause the display by pressing the space bar (press space again to restart the display).

Other Menu Options

Two more menu selections let you save programs and load them back into the computer. These are SAVE FILE (SAVE DATA in the 64 version) and LOAD FILE; their operation is quite straightforward. When you press S or L, MLX asks you for the filename. The 64 version will follow this by asking you to press either D or T to select disk or tape.

Those using the 64 version will notice the disk drive starting and stopping several times during a load or save. Don't panic; this is normal behavior. MLX opens and reads from or writes to the file instead of using the usual LOAD and SAVE commands. Disk users should also note that the drive prefix 0: is automatically added to the filename (line 750), so this should *not* be included when entering the name. (This also precludes the use of @ for Save-with-Replace, so remember to give each version you save a different name.)

Remember that MLX saves the entire workspace area from the starting address to the ending address, so the save or load may take longer than you might expect if you've entered only a small amount of data from a long listing. When saving a partially completed listing, make sure to note the address where you stopped typing so you'll know where to resume entry when you reload.

MLX reports any errors detected during the save or load. For the 64 version, the standard disk or tape error messages will be displayed. (Tape users should bear in mind that the Commodore 64 is never able to detect errors when saving to tape.) The 64 version also has three special load error messages: INCORRECT STARTING ADDRESS, which means the file you're trying to load does not have the starting address you specified when you ran MLX; LOAD ENDED AT *address*, which means the file you're trying to load ends before the ending address you specified when you started MLX; and TRUNCATED AT ENDING ADDRESS, which means the file you're trying to load extends beyond the ending address you specified when you started MLX. If you see one of these messages and feel certain that you've loaded the right file, exit and rerun MLX, being careful to enter the correct starting and ending addresses.

The Apple version simply displays the message DISK ERROR if a problem is detected during a Save or Load. If you're not sure why a disk error has occurred, check the drive. Make sure there's a formatted disk in the drive and that it was formatted by the same operating system you're using for MLX (ProDOS or DOS 3.3). If you're trying to save a file and see an error message, the disk might be full. Either save the file on another disk or quit MLX (by pressing the Q key), delete an old file or two, then run MLX again. Your typing should still be safe in memory. If the error message appears during a Load, you may have specified a filename that doesn't exist on the disk.

The Quit menu option has the obvious effect—it stops MLX and enters BASIC. In the 64 version the RUN/STOP key is disabled, so the Q option lets you exit the program without turning off the computer. (Of course, RUN/STOP–RESTORE for the 64 or CONTROL-RESET for the Apple also gets you out.) The 64 version will ask for verification; press Y to exit to BASIC, or any other key to return to the menu. After quitting, you can type RUN again and reenter MLX without losing your data, as long as you don't use the clear workspace option in 64 MLX.

The Finished Product

When you've finished typing all the data for an ML program and saved your work, you're ready to see the results. The instructions for loading and using the finished product vary from program to program. Some Commodore 64 ML programs are designed to be loaded and run like BASIC programs, so all you need to type is LOAD *"filename"*,8 for disk or LOAD *"filename"* for tape, and then RUN. (Such programs usually have 0801 as their MLX starting address.) Others must be reloaded to specific addresses with a command such as LOAD *"filename"*,8,1 for disk or LOAD *"filename"*,1,1 for tape, then started with a SYS to a particular memory address. (On the Commodore 64, the most common starting address for such programs is 49152, which corresponds to MLX address C000.) In either case, you should always refer to the article which accompanies the ML listing for information on loading and running the program. For the Apple, you need to BRUN the program, or you may BLOAD and start the program with a CALL. Again, refer to the article accompanying the machine language program for instructions.

An Ounce Of Prevention

By the time you finish typing in the data for a long ML program, you'll have several hours invested in the project. Don't take chances—use our "Automatic Proofreader" to type the new MLX, and then test your copy *thoroughly* before first using it to enter any significant amount of data. Make sure all the menu options work as they should. Enter fragments of the program starting at several different addresses, then use the Display option to verify that the data has been entered correctly. And be sure to test the Save and Load options several times to ensure that you can recall your work from disk or tape. Don't let a simple typing error in the new MLX cost you several nights of hard work.

In the Apple version, line 100 traps all errors to line 610. If MLX is typed in correctly, then only disk errors should normally be encountered. A disk error

message when you're not trying to access the drive—for example, when you first start entering data—indicates a typing error in the MLX program itself. If this occurs, hit CONTROL-RESET to break out of MLX and carefully compare your entry against the printed listing.

For instructions on entering these listings, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

Program 1: MLX For Commodore 64

Version by Ottis Cowper, Technical Editor

```
100 POKE 56,50:CLR:DIM IN$,I,J
    ,A,B,A$,B$,A(7),N$ :rem 34
110 C4=48:C6=16:C7=7:Z2=2:Z4=2
    54:Z5=255:Z6=256:Z7=127
                       :rem 238
120 FA=PEEK(45)+Z6*PEEK(46):BS
    =PEEK(55)+Z6*PEEK(56):H$="
    01234567B9ABCDEF" :rem 118
130 R$=CHR$(13):L$="{LEFT}":S$
    =" ":D$=CHR$(20):Z$=CHR$(0
    ):T$="{13 RIGHT}" :rem 173
140 SD=54272:FOR I=SD TO SD+23
    :POKE I,0:NEXT:POKE SD+24,
    15:POKE 788,52     :rem 194
150 PRINT"{CLR}"CHR$(142)CHR$(
    8):POKE 53280,15:POKE 5328
    1,15               :rem 104
160 PRINT T$" {RED}{RVS}
    {2 SPACES}[8 @][2 SPACES}"
    SPC(28)"{2 SPACES}{OFF}
    {BLU} MLX II {RED}{RVS}
    {2 SPACES}"SPC(28)"
    {)2 SPACES}{BLU}" :rem 121
170 PRINT"{3 DOWN}{3 SPACES}CO
    MPUTE!'S MACHINE LANGUAGE
    {SPACE}EDITOR{3 DOWN}"
                       :rem 135
180 PRINT"{BLK}STARTING ADDRES
    S[4]";:GOSUB300:SA=AD:GOSU
    B1040:IF F THEN180:rem 113
190 PRINT"{BLK}{2 SPACES}ENDIN
    G ADDRESS[4]";:GOSUB300:EA
    =AD:GOSUB1030:IF F THEN190
                       :rem 173
200 INPUT"{3 DOWN}{BLK}CLEAR W
    ORKSPACE [Y/N][4]";A$:IF L
    EFT$(A$,1)<>"Y"THEN220
                        :rem 9
210 PRINT"{2 DOWN}{BLU}WORKING
    ...";:FORI=BS TO BS+EA-SA+
    7:POKE I,0:NEXT:PRINT"DONE
    "                  :rem 139
220 PRINTTAB(10)"{2 DOWN}{BLK}
    {RVS} MLX COMMAND MENU
    {DOWN}[4]":PRINT T$"{RVS}E
    {OFF}NTER DATA"    :rem 62
230 PRINT T$"{RVS}D{OFF}ISPLAY
     DATA":PRINT T$"{RVS}L
    {OFF}OAD DATA"     :rem 19
240 PRINT T$"{RVS}S{OFF}AVE FI
    LE":PRINT T$"{RVS}Q{OFF}UI
    T{2 DOWN}{BLK}"    :rem 238
250 GET A$:IF A$=N$ THEN250
                       :rem 127
260 A=0:FOR I=1 TO 5:IF A$=MID
    $("EDLSQ",I,1)THEN A=I:I=5
                       :rem 42
270 NEXT:ON A GOTO420,610,690,
    700,280:GOSUB1060:GOTO250
                       :rem 97
280 PRINT"{RVS} QUIT ":INPUT"
    {DOWN}[4]ARE YOU SURE [Y/N
    ]";A$:IF LEFT$(A$,1)<>"Y"T
    HEN220             :rem 189
290 POKE SD+24,0:END   :rem 95
300 IN$=N$:AD=0:INPUTIN$:IFLEN
    (IN$)<>4THENRETURN :rem 31
310 B$=IN$:GOSUB320:AD=A:B$=MI
    D$(IN$,3):GOSUB320:AD=AD*2
    56+A:RETURN        :rem 225
320 A=0:FOR J=1 TO 2:A$=MID$(B
    $,J,1):B=ASC(A$)-C4+(A$>"@
    ")*C7:A=A*C6+B     :rem 143
330 IF B<0 OR B>15 THEN AD=0:A
    =-1:J=2            :rem 132
340 NEXT:RETURN        :rem 240
350 B=INT(A/C6):PRINT MID$(H$,
    B+1,1);:B=A-B*C6:PRINT MID
    $(H$,B+1,1);:RETURN:rem 42
360 A=INT(AD/Z6):GOSUB350:A=AD
    -A*Z6:GOSUB350:PRINT":";
                       :rem 32
370 CK=INT(AD/Z6):CK=AD-Z4*CK+
    Z5*(CK>Z7):GOTO390:rem 131
380 CK=CK*Z2+Z5*(CK>Z7)+A
                       :rem 168
390 CK=CK+Z5*(CK>Z5):RETURN
                       :rem 159
400 PRINT"{DOWN}STARTING AT[4]
    ";:GOSUB300:IF IN$<>N$ THE
    N GOSUB1030:IF F THEN400
                       :rem 75
410 RETURN             :rem 117
420 PRINT"{RVS} ENTER DATA ":G
    OSUB400:IF IN$=N$ THEN220
                       :rem 85
430 OPEN3,3:PRINT      :rem 34
440 POKE198,0:GOSUB360:IF F TH
    EN PRINT IN$:PRINT"{UP}
    {5 RIGHT}";            :rem 6
450 FOR I=0 TO 24 STEP 3:B$=S$
    :FOR J=1 TO 2:IF F THEN B$
    =MID$(IN$,I+J,1)   :rem 226
460 PRINT"{RVS}"B$L$;:IF I<24T
    HEN PRINT"{OFF}";  :rem 15
470 GET A$:IF A$=N$ THEN470
                       :rem 135
480 IF(A$>"/"ANDA$<":")OR(A$>"
    @"ANDA$<"G")THEN540
                       :rem 100
490 IF A$=R$ AND((I=0)AND(J=1)
    OR F)THEN PRINT B$;:J=2:NE
    XT:I=24:GOTO550    :rem 46
500 IF A$="{HOME}" THEN PRINT
    {SPACE}B$:J=2:NEXT:I=24:NE
    XT:F=0:GOTO440     :rem 66
510 IF(A$="{RIGHT}")ANDF THENP
    RINT B$L$;:GOTO540:rem 107
520 IF A$<>L$ AND A$<>D$ OR((I
    =0)AND(J=1))THEN GOSUB1060
    :GOTO470           :rem 232
530 A$=L$+S$+L$:PRINT B$L$;:J=
    2-J:IF J THEN PRINT L$;:I=
    I-3                :rem 12
540 PRINT A$;:NEXT J:PRINT S$;
                       :rem 2
550 NEXT I:PRINT:PRINT"{UP}
    {5 RIGHT}";:INPUT#3,IN$:IF
     IN$=N$ THEN CLOSE3:GOTO22
    0                  :rem 106
560 FOR I=1 TO 25 STEP3:B$=MID
    $(IN$,I):GOSUB320:IF I<25
    {SPACE}THEN GOSUB380:A(I/3
    )=A                :rem B1
570 NEXT:IF A<>CK THEN GOSUB10
    60:PRINT"{BLK}{RVS} ERROR:
     REENTER LINE [4]":F=1:GOT
    O440               :rem 161
580 GOSUB1080:B=BS+AD-SA:FOR I
    =0 TO 7:POKE B+I,A(I):NEXT
                       :rem 245
590 AD=AD+8:IF AD>EA THEN CLOS
    E3:PRINT"{DOWN}{BLU}** END
     OF ENTRY **{BLK}{2 DOWN}"
    :GOTO700           :rem 207
600 F=0:GOTO440        :rem 84
610 PRINT"{CLR}{DOWN}{RVS} DIS
    PLAY DATA ":GOSUB400:IF IN
    $=N$ THEN220       :rem 146
620 PRINT"{DOWN}{BLU}PRESS:
    {RVS}SPACE{OFF} TO PAUSE,
    {SPACE}{RVS}RETURN{OFF} TO
     BREAK[4]{DOWN}"   :rem 241
630 GOSUB360:B=BS+AD-SA:FORI=B
    TO B+7:A=PEEK(I):GOSUB350:
    GOSUB380:PRINT S$; :rem 56
640 NEXT:PRINT"{RVS}";:A=CK:GO
    SUB350:PRINT       :rem 144
650 F=1:AD=AD+8:IF AD>EA THENP
    RINT"{DOWN}{BLU}** END OF
    {SPACE}DATA **":GOTO220
                       :rem 170
660 GET A$:IF A$=R$ THEN GOSUB
    1080:GOTO220       :rem 65
670 IF A$=S$ THEN F=F+1:GOSUB1
    080                :rem 28
680 ONFGOTO630,660,630:rem 224
690 PRINT"{DOWN}{RVS} LOAD DAT
    A ":OP=1:GOTO710   :rem 31
700 PRINT"{DOWN}{RVS} SAVE FIL
    E ":OP=0           :rem 32
710 IN$=N$:INPUT"{DOWN}FILENAM
    E[4]";IN$:IF IN$=N$ THEN22
    0                  :rem 229
720 F=0:PRINT"{DOWN}{BLK}{RVS}
    T{OFF}APE OR {RVS}D{OFF}IS
    K: [4]";           :rem 66
730 GET A$:IF A$="T"THEN PRINT
    "T{DOWN}":GOTO8B0  :rem 90
740 IF A$<>"D"THEN730  :rem 90
750 PRINT"D{DOWN}":OPEN15,8,15
    ,"I0:":B=EA-SA:IN$="0:"+IN
    $:IF OP THEN810    :rem 163
760 OPEN 1,8,8,IN$+",P,W":GOSU
    BB60:IF A THEN220  :rem 66
770 AH=INT(SA/256):AL=SA-(AH*2
    56):PRINT#1,CHR$(AL);CHR$(
    AH);               :rem 221
780 FOR I=0 TO B:PRINT#1,CHR$(
    PEEK(BS+I));:IF ST THEN800
                       :rem 171
790 NEXT:CLOSE1:CLOSE15:GOTO94
    0                  :rem 230
800 GOSUB1060:PRINT"{DOWN}
    {BLK}ERROR DURING SAVE:[4]
    ":GOSUB860:GOTO220 :rem 61
B10 OPEN 1,8,8,IN$+",P,R":GOSU
    B860:IF A THEN220  :rem 57
820 GET#1,A$,B$:AD=ASC(A$+Z$)+
    256*ASC(B$+Z$):IF AD<>SA T
    HEN F=1:GOTO850    :rem 155
B30 FOR I=0 TO B:GET#1,A$:POKE
     BS+I,ASC(A$+Z$):IF ST AND
    (I<>B)THEN F=2:AD=I:I=B
                       :rem 180
840 NEXT:IF ST<>64 THEN F=3
                       :rem 20
850 CLOSE1:CLOSE15:ON ABS(F>0)
    +1 GOTO960,970     :rem 12
860 INPUT#15,A,A$:IF A THEN CL
    OSE1:CLOSE15:GOSUB1060:PRI
    NT"{RVS}ERROR: "A$:rem 114
870 RETURN             :rem 127
880 POKE183,PEEK(FA+2):POKE187
    ,PEEK(FA+3):POKE188,PEEK(F
    A+4):IFOP=0THEN920:rem 178
890 SYS 63466:IF(PEEK(783)AND1
    )THEN GOSUB1060:PRINT"
    {DOWN}{RVS} FILE NOT FOUND
```

```
    ":GOTO690          :rem 34
900 AO=PEEK(829)+256*PEEK(830)
    :IF AO<>SA THEN F=1:GOTO97
    0                  :rem 201
910 A=PEEK(831)+256*PEEK(832)-
    1:F=F-2*(A<EA)-3*(A>EA):AD
    =A-AD:GOTO930      :rem 75
920 A=SA:B=EA+1:GOSUB1010:POKE
    780,3:SYS 63338    :rem 107
930 A=BS:B=BS+(EA-SA)+1:GOSUB1
    010:ON OP GOTO950:SYS 6359
    1                  :rem 38
940 GOSUB1080:PRINT"{BLU}** SA
    VE COMPLETEO **":GOTO220
                       :rem 139
950 POKE147,0:SYS 63562:IF ST<
    >64 THEN970        :rem 39
960 GOSUB1080:PRINT"{BLU}** LO
    AD COMPLETEO **":GOTO220
                       :rem 126
970 GOSUB1060:PRINT"{BLK}{RVS}
    ERROR DURING LOAO:{DOWN}
    {4}":ON F GOSUB980,990,100
    0:GOTO220          :rem 233
980 PRINT"INCORRECT STARTING A
    DDRESS (";:GOSUB360:PRINT"
    )":RETURN          :rem 145
990 PRINT"LOAO ENOED AT ";:AD=
    SA+AD:GOSUB360:PRINT O$:RE
    TURN               :rem 159
1000 PRINT"TRUNCATEO AT ENOING
     ADORESS":RETURN :rem 166
1010 AH=INT(A/256):AL=A-(AH*25
    6):POKE193,AL:POKE194,AH
                       :rem 95
1020 AH=INT(B/256):AL=B-(AH*25
    6):POKE174,AL:POKE175,AH:
    RETURN             :rem 122
1030 IF AD<SA OR AO>EA THEN105
    0                  :rem 135
1040 IF(AO>511 AND AD<40960)OR
    (AD>49151 ANO AO<53248)TH
    EN GOSUB1080:F=0:RETURN
                       :rem 104
1050 GOSUB1060:PRINT"{RVS} INV
    ALID ADDRESS {OOWN}{BLK}"
    :F=1:RETURN        :rem 224
1060 POKE SO+5,31:POKE SD+6,20
    8:POKE SD,240:POKE SO+1,4
    :POKE SD+4,33      :rem 19
1070 FOR S=1 TO 100:NEXT:GOTO1
    090                :rem 90
10B0 POKE SD+5,8:POKE SO+6,240
    :POKE SD,0:POKE SD+1,90:P
    OKE SD+4,17        :rem 182
1090 FOR S=1 TO 100:NEXT:POKE
    {SPACE}SD+4,0:POKE SO,0:P
    OKE SO+1,0:RETURN  :rem 8
```

Program 2: MLX For Apple

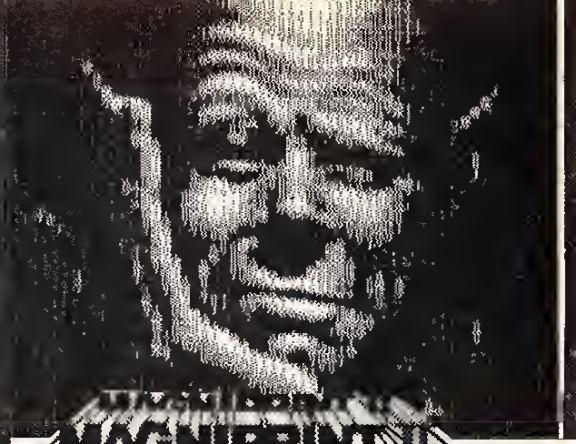
Version by Tim Victor, Editorial Programmer

```
100 N = 9: HOME : NORMAL : PRIN
    T "APPLE MLX": POKE 34,2: O
    NERR  GOTO 610
110 VTAB 1: HTAB 20: PRINT "STA
    RT ADDRESS";: GOSUB 530: IF
     A = 0 THEN  PRINT  CHR$ (7
    ): GOTO 110
120 S = A
130 VTAB 2: HTAB 20: PRINT "END
     ADDRESS  ";: GOSUB 530: IF
     S >  = A OR A = 0 THEN  PR
    INT  CHR$ (7): GOTO 130
140 E = A
150 PRINT : PRINT "CHOOSE: (E)NT
    ER DATA";: HTAB 22: PRINT "
    (D)ISPLAY DATA": HTAB 8: PR
    INT "(L)OAD FILE  (S)AVE FI
    LE  (Q)UIT": PRINT
160 GET A$: FOR I = 1 TO 5: IF
    A$ < > MID$ ("EDLSQ",I,1) T
    HEN  NEXT : GOTO 160
170 ON I GOTO 270,220,180,200:
    POKE 34,0: END
180 INPUT "FILENAME: ";A$: IF A
    $ <  > "" THEN  PRINT  CHR$
     (4);"BLOAD";A$;",A";S
190 GOTO 150
200 INPUT "FILENAME: ";A$: IF A
    $ <  > "" THEN  PRINT  CHR$
     (4);"BSAVE";A$;",A";S;",L"
    ;E - S
210 GOTO 150
220 GOSUB 590: IF B = 0 THEN 15
    0
230 FOR B = B TO E STEP B:L = 4
    :A = B: GOSUB 580: PRINT A$
    ;": ";:L = 2
240 FOR F = 0 TO 7:V(F + 1) = P
    EEK (B + F): NEXT : GOSUB 5
    60:V(9) = C
250 FOR F = 1 TO N:A = V(F): GO
    SUB 580: PRINT A$" ";: NEXT
     : PRINT : IF  PEEK (49152)
    < 128 THEN  NEXT
260 POKE 49168,0: GOTO 150
270 GOSUB 590: IF B = 0 THEN 15
    0
280 FOR B = B TO E STEP B
290 HTAB 1:A = B:L = 4: GOSUB 5
    80: PRINT A$;": ";: CALL 64
    668:A$ = "":P = 0: GOSUB 33
    0: IF L = 0 THEN 150
300 GOSUB 470: IF F < > N THEN
    PRINT CHR$ (7);: GOTO 290
310 IF N = 9 THEN GOSUB 560: IF
     C < > V(9) THEN PRINT CHR$
     (7);: GOTO 290
320 FOR F = 1 TO B: POKE B + F
    - I,V(F): NEXT : PRINT : NE
    XT : GOTO 150
330 IF LEN (A$) = 33 THEN A$ =
    O$:P = 0: PRINT CHR$ (7);
340 L = LEN (A$):O$ = A$:O = P:
    L$ = "": IF P > 0 THEN L$ =
      LEFT$ (A$,P)
350 R$ = "": IF P < L - I THEN
    R$ =  RIGHT$ (A$,L - P - 1)
360 HTAB 7: PRINT L$;: FLASH :
    IF P < L THEN PRINT MID$ (A
    $,P + 1,1);: NORMAL : PRINT
     R$;
370 PRINT " ";: NORMAL
380 K = PEEK (49152): IF K < 12
    8 THEN 380
390 POKE 49168,0:K = K - 128
400 IF K = 13 THEN HTAB 7: PRIN
    T A$;" ";: RETURN
410 IF K = 32 OR K > 47 AND K <
     58 OR K > 64 AND K < 71 TH
    EN A$ = L$ + CHR$ (K) + R$:
    P = P + 1
420 IF K = 4 THEN A$ = L$ + R$
430 IF K = 9 THEN A$ = L$ + " "
     +  MID$ (A$,P + 1,1) + R$
440 IF K = 8 THEN P = P - (P >
    0)
450 IF K = 21 THEN P = P + (P <
     L)
460 GOTO 330
470 F = 1:D = 0: FOR P = 1 TO L
    EN (A$):C$ = MID$ (A$,P,1):
     IF C$ > "/" AND C$ < ":" TH
    EN  RETURN
480 IF C$ < > " " THEN  GOSUB 5
    20:V(F) = J + 16 * (D = 1)
    * V(F):D = D + 1
490 IF D > 0 AND C$ = " " OR D
    = 2 THEN D = 0:F = F + 1
500 NEXT : IF D = 0 THEN F = F
    - 1
510 RETURN
520 J = ASC (C$):J = J - 48 - 7
     * (J > 64): RETURN
530 A = 0: INPUT A$:A$ = LEFT$
    (A$,4): IF LEN (A$) = 0 THE
    N RETURN
540 FOR P = 1 TO LEN (A$):C$ =
    MID$ (A$,P,1): IF C$ < "0"
    OR C$ > "9" AND C$ < "A" OR
     C$ > "Z" THEN A = 0: RETUR
    N
550 GOSUB 520:A = A * 16 + J: N
    EXT : RETURN
560 C = INT (B / 256):C = B - 2
    56 * C - 255 * (C > 127):C
    = C - 255 * (C > 255)
570 FOR F = 1 TO B:C = C * 2 -
    255 * (C > 127) + V(F):C =
    C - 255 * (C > 255): NEXT :
     RETURN
580 I = FRE (0):A$ = "": FOR I
    = 1 TO L:T =  INT (A / 16):
    A$ =  MID$ ("0123456789ABCD
    EF",A - 16 * T + 1,1) + A$:
    A = T: NEXT : RETURN
590 PRINT "FROM ADDRESS ";: GOS
    UB 530: IF S > A OR E < A O
    R A = 0 THEN B = 0: RETURN
600 B = S + B * INT ((A - S) /
    B): RETURN
610 PRINT "DISK ERROR": GOTO 15
    0
```


Attention Programmers

COMPUTE! magazine is currently laaking far quality articles on Cammadare, Atari, Apple, and IBM computers (including the Cammadare Amiga and Atari ST). If yau have an interesting hame application, educatianal pragram, pragramming utility, ar game, submit it ta COMPUTE!, P.O. Box 5406, Greensbara, NC 27403. Or write far a capy af our "Writer's Guidelines."

NEW FOR YOUR ATARI®



Digitize your picture with the perfect hardware/software combination: COMPUTER EYES and MAGNIPRINT II+

Digitize your voice with Parrot

Computer Eyes lets you take any form of video input and saves it as a high-resolution graphics screen. You can use a video camera, VCR TV output, video disk, other computers, etc. Now you can capture your picture, your friends or any video image and show it on an Atari computer. Computer Eyes is an innovative slow scan device that connects between any standard video source and your Atari computer (see the review in A.N.A.L.O.G. magazine)

- Do a complete Hi-Res scan in under 6 seconds
- Unique multi-scan mode provides realistic grey scale images in 24 seconds and up for more detail scans
- Full one-year warranty on parts and labor
- Plugs into your Atari joystick ports and uses a standard video phono plug

Now anyone can create the kind of graphics seen in this ad. When Computer Eyes is combined with Magniprint II+ you get unique capabilities that no other system can offer

- Print your pictures in up to 19 different sizes, including poster sizes several feet across
- Print pictures with full shading for a level of realism even better than your TV screen
- Take your Computer Eyes images and modify them with your Koala Pad, Atarr Touch Tablet, Micro Illustrator program or Magniprint's special touch up feature
- Works with Epson, Gemini, Panasonic, NEC, Citoh and other compatible dot-matrix printers (B50 interface or equivalent required)
- Supports all Magniprint II+ features

COMPUTER EYES, captura softwara and MAGNIPRINT II+ ..... Only $114.95

COMPUTER EYES/MAGNIPRINT Camera System
A complete ready to run system for those without access to video equipment. This system includes Computer Eyes, Magniprint II+, a high quality B/W video camera, and a 10 ft coaxial cable with appropriate connectors. Only **$299.95**

COMPUTER EYES alone (with capture and display software only) **$99.95**

Computer Eyes Gr.9 Acquisition Software **$12**

MAGNIPRINT II+

Magniprint II is a versatile and powerful program which will meet most if not all your print out needs" ANALOG MAGAZINE

Print your Atari® graphics like you've never seen before!

The plus is in the shading — now Magniprint lets you print the highest quality reproductions of your screens available anywhere. It lets you choose the shading you want for the different colors and shades on the screen. It allows you to adjust the shades while your picture is displayed on the screen, so what you see is what you get. Magniprint prints the shades on your standard printer and paper by using specially designed dot patterns, yielding better looking output than you get on your screen.

This new feature is perfect for printing pictures from Koala Pad, Touch Tablet, Micro Illustrator and others. It will amaze you when you print pictures from Computer Eyes or Strip Poker.

Now get your pictures shaded the way you want them

Magniprint II is the most powerful print utility available for your computer! It is the only software that can accurately print GTIA modes with 16 shades.

Pictures from PRINTSHOP*. Prints 19 different sizes from ½ page to giant wall sized posters. Prints across or down the page, regular or inverse, with a centering option. Works with printers compatible with NEC, CITOH, EPSON, & Gemini (B50 interface or equivalent required). Accepts your own graphics creations or those created with Graphics Master*, Micropainter*, Fun With Art*, SYN_Graph*, Atari Graphics Light Pen*, B/Graph*, Koala Pad*, Atari Touch Tablet*, Paint*, Strip Poker*, Graphics Magician*, and others. • Includes many beautiful sample pictures. • Does amazing graphics with your standard printer and paper. • Special feature lets you modify pictures with your joystick. • Add text to your pictures. • Can convert a graphics mode 8 picture to a graphics mode 8 one (and vice versa). • Allows you to print the whole screen or any selected portion of the screen, in multiple sizes. • Includes HELP screen for easy use. • Prints GTIA modes 9, 10, and 11 with 16 different shades. Posters print along continuous sheets of paper which are attached together when done.

40% Faster ONLY **$24.95**

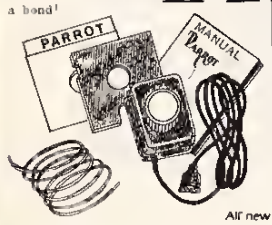
NOW prints text modes **0,1,2** with full Magniprint capabilities

FREE With every Magniprint II order get "PRINTALL" FREE. PRINTALL prints your programs and files just as they appear on the screen. It clearly prints INVERSE and all the Atari graphics characters, and prints in regular, condensed print. This alone is worth the price.

DIGITIZE YOUR VOICE

PARROT

PARROT... So good, it is being used by professional musicians in a band!



All new sound digitalizer and synthesizer for your Atari. Tired of low-quality mechanical sounding voice output? Now you can make any Atari speak in your own voice. Tired of lousy tone sound? Now any Atari can play a whole orchestra complete with a singing choir. "The Parrot" digital sound synthesizer system lets you do all this and much more.

How it works — "The Parrot" system plugs into your joystick port and lets you record pure digital sound from your stereo, TV, microphone, or any other sound source. The special Parrot software lets you play back this high quality sound on any Atari system with no special hardware needed. It even lets you put this unbelievable sound right into your own programs that will run on anyone's Atari. It also includes digital sequencer software that lets you turn your Atari into a synthesizer comparable to those costing thousands of dollars. Turn any natural sound into a musical instrument, or design your own custom sounds. Imagine playing a song with the sounds of a dog's bark, a chinese gong, a car's honk, your own voice, or anything your imagination can come up with. It turns your keyboard into an organ and lets you instantly switch between up to nine different digital sounds, each with three full octaves of notes. Recording time varies depending on available memory and quality level desired. You've got to hear it to believe it!!!!

THE PARROT digital input hardware and playback/synthesizer software with sample sounds and demos

All For Only **$39.95**

AT LAST A UTILITY THAT DOES IT ALL. Scans & Analyses ALL programs from Atari® programs stored on Disk, Cartridge or directly from memory. • Converts complex machine language into readable assembler. • Transforms ANY Atari® BASIC program into listable, modifyable BASIC. • Changes 8 K 8 or 16K cartridge into a binary load file and source file that you can view & change using regular Atari® assembler. Clearly shows protection techniques such as BAD SECTORS, BAD DATA MARKS, DUPLICATE SECTORS and FORCED CRC ERRORS.

FEATURES!

THE Scanalyzer

NEW AND IMPROVED

NEW FEATURES Now fully supports DOS 2.5 and includes one pass sector copier for 130XE owners. Even finds and displays hidden directories. No other program can do this!

$29.95

Complete with instructions in theory and use

IMPERSONATOR

CARTRIDGE TO DISK COPY SYSTEM

CARTRIDGE TO DISK COPY SYSTEM Yes, for only $29.95, you can make working copies of all your Atarr computer cartridges (16K or less). Our special package will let you save your cartridges to ordinary disk files. They will run exactly like the originals when used with the Impersonator. Each disk holds up to 12 cartridge programs. Now you can put all your real cartridges away for safe keeping and use the Impersonator for everything. YES, IT REALLY WORKS. The Impersonator does everything the high-priced cartridge back-up systems do — and more. **ONLY $29.95**

SCHEMATIC DESIGNER

Schematic Designer is designed to help you create the electronic gadget of your dreams. Features 28 built-in symbols and the possibility of hundreds more, in high resolution graphics. You can add titles, zoom in and print out your schematics. SCHEMATIC DESIGNER includes many sample schematics which can be used to build many projects, including a working light pen.

SCHEMATIC DESIGNER ON A DISK... Only $24.95

BOOK I + DISK (The Original) Thoroughly explains the techniques used by advanced software pirates and the copy protection methods used to stop them. It offers clear and understandable explanations sophisticated enough for software writers of any scale yet easy enough for a beginner just wanting to learn more about Atari® computers. A MUST READ FOR ALL ATARI® OWNERS.

BOOK INCLUDES • Duplicate sectoring • Custom disk formatting • Creating "BAD" sectors • Hardware data keys • Legal protection like copyrights, trade secrets, patents • Protecting BASIC programs • Self-modifying Code • ROM + EPSOM cartridges • Hidden serial numbers • Self-destructing programs • Freeware • Misassigned sectoring • Much, much more

DISK INCLUDES • Directory mover • VTOC scanner • Duplicate sector finder • Sector mover • Bad sector writer • Sector data displayer • Autorun builder • Other useful programs

This comprehensive book and disk package should not be confused with low quality imitations offered elsewhere.

BOOK II + DISK II. Advanced Software Protection. This all new sequel starts where the highly acclaimed Book I leaves off. Book II is the most up-to-date resource available for the Atari® owner. Includes reviews and explanations of products such as The Happy Enhancement* The Impossible* The Scanalyzer* The Chip* The Pill* and Super Pill* & many others.

Book II Tells you specifically what they copy, what they won't, how they are used, and the details of how they work. Book II also includes such topics as • Transmitting protected programs • Copying disks with more than 19 sectors/track. Includes the newest protection methods by companies like Synapse* AND Electronic Arts* • Data encryption • Phreaking methods • Program worms • Logic bombs • Bank select cartridges • Random access codes • New trends in software law • Sample BASIC + Assembler programs • On-line security • And much more.

DISK II INCLUDES. • Automatic program protector • Custom format detector • Newest protection demos • Forced password appender • Data encrypter • And much more.

Book + Disk Packages only $24.95 each or Special Offer both for only $39.95

ALPHA SYSTEMS

MAIL TO **ALPHA SYSTEMS - Dept C**
4435 Maplepark Rd / Stow, OH 44224 Send check or money order. Include $2.00 shp. & hdlg. Ohio residents 5½% sales tax. CALL 216 374 7469 to charge to MasterCard or VISA

BONUS Order any 3 programs and get FREE your choice
☐ Deluxe Space Games (3 games on a disk) or ☐ Disk Pak 1000 (Utility Package)

All for your Atari Computers. Disk drive and 48K required. Atari® is a registered trademark of Atari Corporation.
*Denotes products not related to Alpha Systems

ALL PRODUCTS WORK WITH ALL ATARI 8 BIT COMPUTER SYSTEMS

ATARI 130XE

ATARI 130XE Super Computer Package

130XE Computer — 1027 Printer
1050 Disk Drive — Atariwriter +

Call for individual & super package price

ATARI PRINTER INTERFACES

Uprint A	54 95	Uprint AW/64K	99 95
Uprint AW/16K	79 95	MPP 1150	54 95

Supra 1000E Modem 44.95
Atari XM-301 Modem 44.95

ATARI 130-XE SUPER PRINTER PACKAGES

SG-10 Printer & U-Print A 269
Panasonic 1091 & U-Print A 299

Super Printer Packages have no extra shipping charges or credit card surcharges when shipped in Continental USA

ATARI 130XE SOFTWARE

BRODERBUND

Print Shop	28 95
Karateka	20 95
Print Shop Graph I, II or III	19 95
Print Shop Comp	27 95

INFOCOM

See Commodore 64 section for items and prices

ELECTRONIC ARTS

Archon II	24 95
Archon	19 95
Seven Cit of Gold	24 95
Skyfox	24 95
Pinball Const	19 95
One on One	24 95
Super Boulder Dash	19 95
Chessmaster 2000	27 95
Racing Destruction	24 95

MICROPROSE

Silent Service	23 95
Gunship	23 95
Acrojet	23 95
F-15 Strike Eagle	23 95
Kennedy Approach	23 95

SSI

See Commodore 64 section for items and prices

MISCELLANEOUS 130XE

Mind Pursuit	23 95
Never Ending Story	23 95
Fooblitzky	27 95
Syncalc	32 95
Typesetter	27 95
Ultima III	37 95
Ultima IV	41 95
Hacker	19 95
Beachead II	27 95
Raid Over Moscow	27 95
Fight Night	23 95
Hardball	23 95
Flight Simulator II	34 95
Letter Perfect	39 95
Data Perfect	39 95
Alternate Reality	27 95
Fleet System II	49 95
Page Designer	23 95
Megafont II	19 95
Rubber Stamp	23 95
Sargon III	34 95
Halley Project	23 95
Synfile	32 95

ATARI 520ST*

Atari 520ST-RGB System ..... Call
Atari 520ST-Monochrome Sys ... Call
SF314DS/DD1 Megabyte Disk Dr ... Call

We warranty all 520ST computers purchased from ComputAbility for ninety days

***Please call for stock availability on Atari ST products before ordering by mail.**

MISCELLANEOUS ST

Gato	34 95	Amazon	34 95
Kings Quest II	34 95	DOS Shell	34 95
ST Talk	17 95	Kissed	27 95
H&D Base	69 95	9 Princess/Amber	34 95
Typesetter ST	27 95	Hacker	29 95
Habadex Phone	39 95	Dragonworld	34 95
Word Invaders	24 95	Treasure Island	27 95
Mean 18 Golf	34 95	Wizard of Oz	27 95
Financial Cookbook	34 95	Transylvania	27 95
BratacCus	34 95	Borrowed Time	34 95
Michtron Utilities	41 95	Mi Term	34 95
Exploding Fist	34 95	Regent Word	34 95
Black Cauldron	27 95	Regent Spell	34 95
Printmaster ST	27 95	Regent Base	Call
PC Intercom	89 95	Goldrunner	27 95
Final Word	94 95	Time Bandit	27 95
Hex	27 95	Zoomracks	59 95
Mudpies	27 95	Easy Draw	109 95
Sundog	27 95	Mindshadow	34 95
Flipside	27 95	Phantasie	27 95
Softspool	27 95	Hippopotamus	Call
VIP Professional	119 95	Supra Hard Drive	Call
Ultima II	39 95	Supra 1200 ST Modem	Call
Perry Mason	34 95	QMI 1200 ST Modem	Call
Oegas	27 95	PC Board Designer	Call
Farenheit 451	34 95	Infocom	See IBM

AMIGA

Call for Hardware and Add-on Peripherals prices

AMIGA SOFTWARE

Hacker	29 95	Maxicom	39 95
Mindshadow	29 95	Maxidesk	49 95
Mastertype	27 95	Maxiplan	119 95
VIP Professional	139 95	One on One	29 95
Transylvania	27 95	7 Cities/Gold	29 95
Crimson Crown	27 95	Skyfox	29 95
Racter	29 95	Marble Madness	29 95
Analyze	69 95	Return/Atlantis	29 95
On line	49 95	Archon	29 95
Artic Fox	29 95	Aegis Software	Call
Bratacca	34 95	Exploding Fist	27 95
Adventure Const	29 95	Deluxe Print	Call
Deluxe Video	Call	Music Studio	39 95
Deluxe Paint	Call	Borrowed Time	29 95
Infocom	See IBM	Financial Cookbook	34 95

APPLE

LAZER 128 APPLE IIC COMPATIBLE

Call for latest prices

APPLE SOFTWARE

BRODERBUND

Print Shop	33 95
Print Shop Graphics I, II or III	17 95
Print Shop Comp	27 95
Karateka	21 95
Carmen Sandiego	25 95
Science Tool Kit	39 95
Bank Street Writer	44 95
Fantavision	34 95

ELECTRONIC ARTS

Adventure Const	34 95
Archon II	27 95
Bard's Tale	29 95
Imagic Football	24 95
Auto-Duel	34 95
Skyfox	27 95
Lords of Conquest	27 95
One on One	27 95
Ultima III	39 95
Ultima IV	39 95
Moebius	39 95
Amnesia	Call

EPYX

Ballblazer	24 95
Winter Games	24 95
Summer Games II	24 95
World's Great/ Football	24 95
Eidolon	24 95
Apshai Trilogy	24 95
Koronis Rift	24 95

INFOCOM

See IBM section for items and prices

MICROPROSE

See Atari 130XE section for items and prices

SIR-TECH

Wizardry/Diam	23 95
Wizardry/Legacy	27 95
Wizardry/Proving	33 95
Wizardry/Wernda	29 95
Wizprint	19 95

SSI

See Commodore 64 section for items & prices

APPLE MISCELLANEOUS

Beachead	23 95
Beachead II	23 95
Gamemaker	34 95
Hacker	27 95
Hardball	24 95
Sundog	27 95
Star League Base	23 95
Microleague Base	29 95
Newsroom	39 95
Clip Art Vol I	20 95
Clip Art Vol II	27 95
Gato	27 95
Kings Quest II	34 95
Dambusters	24 95
Alternate Reality	27 95
Kung Fu Master	25 95
Karate Champ	25 95
Strip Poker	23 95
Math Blaster	34 95
Fight Night	24 95
Crossword Magic	34 95
Typing Tutor III	34 95
Law Of The West	24 95
PSI Trading	24 95
Phantasie II	27 95

IBM PC

Corona PC-480 Compatible ..... Call
Corona Portable PC Compatible ... Call

PC Multifunction Boards

We carry the complete line of AST, Hercules, Paradise, STB, and Quadram

Call for current prices

IBM PC SOFTWARE

BRODERBUND

Print Shop	39 95
Print Shop Graph I	27 95
Bank Street Writer	49 95
Ancient Art of War	29 95

MICROPROSE

F-15 Strike Eagle	23 95
Kennedy Approach	27 95
Acrojet	27 95
Silent Service	27 95

IBM MISCELLANEOUS

Mean 18	34 95
Gato	27 95
Wizardry	39 95
Strip Poker	27 95
Jet	34 95
Newsroom	39 95
Super Sunday	24 95
Home Pak	34 95
Alter Ego	34 95
Hacker	29 95
Borrowed Time	27 95
Isgur Portfolio	164 95
Typing Tutor	34 95

INFOCOM

Deadline	34 95
Starcross	34 95
Zork I	27 95
Zork II or III	29 95
Witness	27 95
Suspended	34 95
Planetfall	27 95
Sorcerer	29 95
Seastalker	27 95
Cutthroats	27 95
Hitchiker	27 95
Suspect	29 95
Wishbringer	27 95
Infidel	29 95
Enchanter	27 95
Spellbreaker	34 95
Mind Forever	34 95
Ballyhoo	27 95

SIERRA

King's Quest	34 95
King's Quest II	34 95
Ultima II	39 95

COMMODORE

COMMODORE 128

C-128 Computer	Call	1670 Modem	159
1571 Disk Drive	Call	1350 Mouse	42 95
1902 Monitor	Call	1750 512K Expander	Call

COMMODORE 128 SOFTWARE

Consultant	41 95	King's Quest II	34 95
Paper Clip/Spell	54 95	Home Pak	34 95
Swiftcalc w/Side	49 95	Superscript 128	59 95
Wordwriter + Spell	49 95	Perfect Writer	49 95
Data Manager II	49 95	Sylvia Porter-128	49 95
Fleet System III	54 95	Jane	39 95
Mach 128	34 95	Matrix	39 95
Superbase 128	69 95	Perfect Filer	49 95

COMMODORE 64 / 128

C-64 Computer	Call	1670 Modem	159
1541 Disk Drive	Call	1802 Monitor	Call
1660 Modem	49 95		

COMMODORE 64/128 SUPER PRINTER PKGS.

SG-10 & Xetec Supergraph 279
Panasonic 1091 & Xetec Supergraph 315
Legend 1080 & Xetec Supergraph 269

Super Printer Packages have no added shipping or charge card surcharges when shipped in Continental USA

While Supplies Last

B.I. 80 Column Card 59.95

XETEC Super Graphic 69.95
Cardco G-Wiz 54.95

COMMODORE 64 SOFTWARE

ACCESS

Mach V Cart	21 95
Leader Board	27 95

SSI

Battle of Antietam	32 95
Fighter Command (No Atari)	37 95
Panzer Grenidier	24 95
USAAF	37 95
Breakthrough/ Ardennes	37 95
Kampgruppe	37 95
Phantasie II (No Atari)	27 95
Broadsides	24 95
Carrier Force	37 95
Comp Ambush	37 95
Field of Fire (No Apple)	24 95
Gemstone Warrior	21 95
Imp Galactum	24 95
Computer Baseball	24 95
Comp Quarterback	24 95
Wizards Crown (No Atari)	27 95
Rings of Zilfin (No Atari)	27 95
Battle Group (No Atari)	39 95

MICROPROSE

See Atari 130XE section for items and prices

INFOCOM

Zork I	24 95
Zork II, or III	27 95
Deadline	29 95
Starcross	29 95
Witness	29 95
Planetfall	24 95
Hitchiker	24 95
Enchanter	24 95
Cutthroats	24 95
Sorcerer	29 95
Spellbreaker	29 95
Ballyhoo	27 95

ELECTRONIC ARTS

Adv Construction	29 95
Mail Order Monster	24 95
Racing Destruction	24 95
Ultima III	37 95
Ultima IV	41 95
Bard's Tale	27 95
Lords of Conquest	27 95
Chessmaster 2000	27 95
Touchdown Ftball	23 95
Ultimate Wizard	24 95
Super Boulderdash	19 95

See Atari 130XE section for rest of items & prices

EPYX

Eidolon	24 95
Koronis Rift	24 95
Winter Games	24 95
Apshai Trilogy	24 95
Fast Load-Cart	24 95
Program/Tool Kit	29 95
Vorpal Utility	22 95
Multiplan	44 95

MISCELLANEOUS COMMODORE 64

Freeze Frame	34 95
Print Shop	28 95
Cal-Kit	34 95
Superbase 64	47 95
Karateka	20 95
Hacker	20 95
Gamemaker	27 95
Ultima II	37 95
Karate Champ	25 95
Sticks of Death	20 95
Kung Fu Master	25 95
Paper Clip/Spell	54 95
Consultant	41 95
Spy vs Spy Vol II	23 95
Internat'l Hockey	19 95
Prt Shop Compan	27 95
Prt Shop Graphics	19 95
Gato	20 95
Jet	29 95
Fleet System II	44 95
Printmaster	24 95
Newsroom	34 95
Fontmaster II	34 95
Sports Lib Vol I	19 95

We stock hundreds of programs for the Apple, Atari, C-64 and IBM

If you don't see it here, don't hesitate to call

GENERAL HARDWARE

Star Micronics Inc — THE POWER BEHIND THE PRINTED WORD.

NEW NX-10

Printer	Call
SG-10	Call
SG-15	369
SD-10	339
SD-15	449
SR-10	489
SR-15	Call

PRINTERS

Panasonic 1091	245
Legend 808	169
Legend 1080	209
Powertype	309
Juki 5510	369
Epson	Call
Panasonic 1080	215

PRINTER BUFFERS

Microfazer	From 169
U-Buff 16K	79 95
U Buff 64K	99 95

MODEMS

Volksmodem 1200	189
Prometheus 1200	299
Maxwell 1200	Call
Maxwell 2400	Call
Prometheus 2400	Call

MONITORS

Thommson RGB	289
Commodore 1802	Call
Commodore 1902A	289
Gold Star	Call
Taxan	Call
Amdek	Call
Panasonic 1300DT	279

ComputAbility™

P.O. Box 17882 Milwaukee WI 53217
ORDER LINES OPEN
Mon-Fri 11 a m - 7 p m CST
Sat 12 p m - 5 p m CST

NO SURCHARGE FOR MASTERCARD & VISA

To Order Call Toll Free
800-558-0003

For Technical Info., Order Inquiries, or for Wisc. Orders
414-351-2007

ORDERING INFORMATION Please specify system For fast delivery send cashier's check or money order Personal and company checks allow 14 business days to clear School P O's welcome **C O D charges are $3 00** In Continental U S A include $3 00 for software orders, 4% shipping for hardware minimum $4 00 Master Card and Visa orders please include card #, expiration date and signature WI residents please include 5% sales tax HI, AK, FPO, APO, Puerto Rico and Canadian orders, please add 5% shipping **minimum $5 00.** All other **foreign orders add 15% shipping, minimum $10 00** All orders shipped outside the Continental U S A are shipped first class insured U S mail If foreign shipping charges exceed the minimum amount, you will be charged the additional amount to get your package to you quickly and safely All goods are new and include factory warranty Due to our low prices all sales are final **All defective returns must have a return authorization number** Please call (414) 351-2007 to obtain an R A # or your return will not be accepted Priced and availability subject to change without notice

Classified

SOFTWARE

FREE APPLE SOFTWARE
Over 1000 Public Domain Programs on 50 diskettes, $5 ea. + $1 s/h per order. Send $1 for catalog (refundable). C&H Enterprizes, Box 29243, Memphis, TN 38127

FREE SOFTWARE CATALOG!
Call Toll-Free 1-800-554-1162, Tevex, Inc. Save 1/3 off retail prices. We carry SSI, Elect. Arts, Infocom, and many more!

LOTTO PICKER. Improve your chances for those Million Dollar Jackpots! Picks LOTTO, WIN-4 and Daily Numbers. All USA & CAN. games incl. Expandable! IBM/C64/TI99 $29.95. Order Now! 1-800-341-1950 ext. 77. Mail Orders: Ridge, 170 B'way, #201C, NY, NY 10038. Catalog.

FREE SOFTWARE for C64, C128, IBM, & CPM. For info send large stamped (39¢) return envelope to: PUBLIC DOMAIN USERS GROUP, PO Box 1442-A1, Orange Park, FL 32067

EXCITING NEW CP/M SOFTWARE PRODUCTS for C64 and C128 computers! Write for info. Madd Programmer Software, 831 32nd Ave., Seattle, WA 98122 (206) 324-6980

Hebrew, Russian, Greek, Korean, Japanese word processors. C64/128. $29.95, IBM PC. $69.95. Also create your own char. sets! Send check to ISRACOMP, POB 1091, K. of Prussia, PA 19406

TI-99/4A QUALITY SOFTWARE for Business, Home and Entertainment. **Bonus Software Offer!** Send for FREE catalog to MICRO-BIZ HAWAII, BOX 1108, PEARL CITY, HI 96782

COMMODORE: TRY BEFORE YOU BUY. Top 25 best-selling games, utilities, new releases. Visa, MasterCard. Free brochure. Rent-A-Disk, 908 9th Ave., Huntington, WV 25701 (304) 522-1665

DISCOUNT SOFTWARE. Amiga/Apple/Atari/ C64-128/IBM PC-PCjr/TRS-80/Timex/Sinclair. Free Catalog: WMJ DATA SYSTEMS, 4 Butterfly Dr., Hauppauge, NY 11788

CHECKBOOK PLUS for ATARI 64K computers. Get the features of expensive electronic checkbooks, not the price. $19.95. A-PLUS, 3790 Blenheim Rd., St. Joseph, MI 49085

Msdicsl/Dentsl Accounts Receivsble
programs for the C64/128, 900 accounts, 1400 transactions. 2 drives and Epson compatible printer req'd. Full featured. $100.00 for program or $20.00 for demo program and manual (credit to purchase) plus $4.50 P&H. Also: Drug Interactions $85.00 (also for IBM PC), California Medi-cal Billing: $150.00, Free info. Peter Bresler, M.D., 2390 Tierra Dr., Los Osos, Calif. 93402 (805) 528-5234

TI-99/4A Software/Hardware bargains. Hard to find items. Huge selection. Fast service. Free catalog. D.E.C., Box 690, Hicksville, NY 11801

ORGANIZE YOUR GENEALOGY with "FAMILY BRANCHES". For C64 & 1541 or 1571 disk drive. Send $19.95 + $2 p/h or SASE for free info to: INVENTURE SOFT, 3353 S. Main, Suite 101, Salt Lake City, UT 84115

DIRECT SALES DISTRIBUTOR'S HELPER - C64
Does inventory, order forms, reports and more. $59.95. Send SASE for free demo kit. RD3 Box 413, Kunkletown, PA 18058

DR. T's MUSIC SOFTWARE
IF YOU CAN USE A WORD PROCESSOR, YOU CAN MAKE MUSIC! MIDI music-composition software for APPLE and COMMODORE computers. Also algorithmic composition programs for computer generation of sequences. Fully editable Bach, Keyboard, Bass and Drum disks. Reviewed in Jan. '85 *COMPUTE!*
(617) 244-6954, Dr. T's, Dept. C, 66 LOUISE RD., CHESTNUT HILL, MA 02167

INTEREST CALCULATIONS.
MAI-2.10 lets your computer help analyze investment decisions. Calc. future value, present value, annuities, sinking funds, loan pymt sched., + more! Menu driven/Simple IBM PC/XT/jr. or compat. Only $49.95 + $7.00 s/h/t. ck/mo. Munier Associates, Inc., Dept. A5, P.O. Box 79314, Houston, Texas 77279 (713) 784-4348

IBM PCjr. OWNERS: We have the software you want. We also have complete 2nd drive additions or kits. FREE CATALOG. OOWL SOFTWARE, 1435 Burnley Sq. N., Columbus, OH 43229

NAME & ADDRESS SYSTEM. IBM PC & XT, $30. Mailing labels + 7 fncs. Free clock pgm with order. Send check to Pratt & Assoc., 218 Orange St., Elgin, IL 61020

AMIGA OWNERS - NEW TOOL FOR STUDENTS: EQUATION PLOTTER generstes snd compsres curves of functions. Find roots, home in on locsl sress...morel 512K exp. RGB. BASIC disk $29.95. Scientific Softwsre, Dept. C, POB 202, Rexford, NY 12148

SHERLOCK'S HOUNDS - Greyhound Anslysis
Use your computer's power and get the winning edge at the track by quickly timing and rating up to 8 dogs per race with Sherlock's exclusive system. Easy to use disk or cassette for Atari (XL or XE), TI-99/4A. For fast, insured delivery send $29.95 to Swift Creek Software, Box 1562 (CP), Cambridge, MA 02238

20% OFF retail price of everything. Send $1.00 for catalog (refundable on 1st order) to: Sonic Software, Box 70, Eagle Lake, MN 56024

HARDWARE

WE CAN SAVE YOU MONEY on items such as computers, stereos, phones and a lot more! Send $2.00 to be added to our mailing list! B&J Computer Sales, P.O. Box 18626, Shreveport, LA 71138

Robot supplies snd sccessories - Mobile platform, DC gear motors, CMOS computers, etc. Call or write for free catalog (707) 224-6000 Intrep, POB 5381, Napa, CA 94581

MISCELLANEOUS

LEARN TO INVEST - STOCKS, BONDS, OPTIONS AND MUTUAL FUNDS. Concise but complete explanation on 82 minute cassette, with printed supplement, by investment professional, Harvard MBA, over 20 years experience. Send $11.40, or write for further information. Berkshire Co., PO Box 3511, Hartford, CT 06103

IBM PCjr. REPORT: THE NATIONAL NEWS-LETTER PCjr.-specific articles, reviews, Public Domain from across the nation. $18/yr. PCjr. CLUB, POB 95067, Schaumburg, IL 60195

THE LOVE LINE
An exclusive on-line BBS system for just one thing: GETTING GUYS AND GALS TOGETHER. For details send SASE to: THE LOVE LINE, R3 BOX 205D, PANA, IL 62557

RIBBONS for MOST PRINTERS at LOW PRICES! DELTA MICRONICS
BOX 10933, ERIE, PA 16514-0933
(814) 455-5667

Esrn dollsrs $$$$$ for the program
YOU write on your computer. Brochure $1. Dalhousie International Software
818 Forest Avenue, Perry, GA 31069

COMPUTE! Clsssified is s low-cost way to tell over 350,000 microcomputer owners about your product or service.

Rates: $25 per line, minimum of four lines. Any or all of the first line set in capital letters at no charge. Add $15 per line for boldface words, or $50 for the entire ad set in boldface (any number of lines).
Terms: Prepayment is required. Check, money order, American Express, Visa, or MasterCard is accepted. Make checks payable to COMPUTE! Publications.
Form: Ads are subject to publisher's approval and must be either typed or legibly printed. One line equals 40 letters and spaces between words. Please underline words to be set in boldface.
General information: Advertisers using post office box numbers in their ads must supply permanent address and telephone numbers. Orders will not be acknowledged. Ad will appear in next available issue after receipt.
Closing: 10th of the third month preceding cover date (e.g., June issue closes March 10th). Send order and remittance to: Harry Blair, Classified Manager, COMPUTE!, P.O. Box 5406, Greensboro, NC 27403. To place an ad by phone, call Harry Blair at (919) 275-9809.
Notice: COMPUTE! Publications cannot be responsible for offers or claims of advertisers, but will attempt to screen out misleading or questionable copy.

Advertisers Index

To Our Readers:

COMPUTE! Publications is a part of the ABC Consumer Magazines group of ABC Publishing, Inc. and recently we consolidated many of our operations and moved our Customer Service Department to the New York ABC headquarters. If you have any questions regarding back issues, disk orders, book orders, or how to place an order, call toll free **1-800-346-6767**. New York residents should call 212-887-8525.

If you want to order a subscription to COMPUTE!, COMPUTE!'s GAZETTE, COMPUTE!'s GAZETTE DISK, or the COMPUTE! DISK, call **1-800-247-5470** or in Iowa call 1-800-532-1272.

Our Editorial Offices remain in Greensboro, North Carolina. If you wish to submit an article for publication, write us at COMPUTE! Publications, Inc., P.O. Box 5406, Greensboro, NC 27403.

We thank you for your interest and continued support of COMPUTE! Publications.

COMPUTE! Publications, Inc.

Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies

Flight Simulator II Scenery Disks

The Challenge of Accomplished Flight

With a realism comparable to (and in some ways even surpassing) $100,000 aircraft flight simulators, Flight Simulator II includes full flight instrumentation and avionics, and provides a full-color out-the-window view. Instruments are arranged in the format standard to modern aircraft. All the radios needed for IFR flight are included. Front, rear, left, right, and diagonal views let you look in any direction. Program features are clearly documented in a 96-page Pilot's Operating Handbook.

For training in proper flight techniques, Flight Simulator II includes another 96-page instruction manual, compiled by two professional flight instructors with over 8,000 hours flight time and 12,000 hours of aviation teaching experience. You'll learn correct FAA-recommended flight procedures, from basic aircraft control through instrument approaches. To reward your accomplishments, the manual even includes a section on aerobatic maneuvers.

The Realism and Beauty of Flight

Go sight-seeing over detailed, realistic United States scenery. High-speed graphic drivers provide an animated out-the-window view in either day, dusk, or night flying modes.

Flight Simulator II features over 80 airports in four different scenery areas: New York, Chicago, Seattle, and Los Angeles. Six additional Scenery Disks covering the entire Western half of the United States are now available in IBM and C64/128 disk formats.

Apple and Atari versions will be released soon. Each disk covers a geographical region of the country in detail, and is very reasonably priced.

The Pure Fun of "World War I Ace"

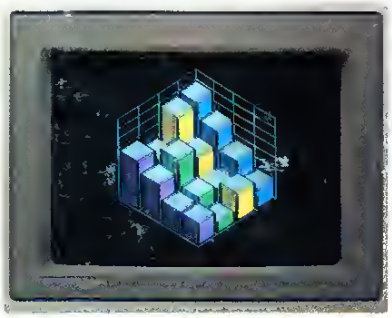
When you think you're ready, you can test your flying skills with the "World War I Ace" aerial battle game. This game sends you on a bombing run over heavily-defended enemy territory. Six enemy fighters will attempt to engage you in combat as soon as war is declared. Your aircraft can carry five bombs, and your machine guns are loaded with 100 rounds of ammunition.

See Your Dealer. Flight Simulator II is available on disk for the Apple II, Atari XL/XE, and Commodore 64/128 computers for $49.95. Scenery Disks for the C64 and IBM PC (Jet or Microsoft Flight Simulator) are $19.95 each. A complete Western U.S. Scenery six-disk set is also available for $99.95. For additional product or ordering information, call (800) 637-4983

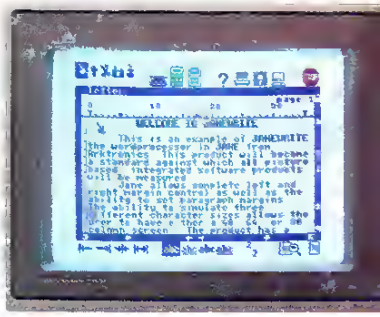
subLOGIC Corporation
713 Edgebrook Drive
Champaign IL 61820
(217) 359-8482 Telex 206995

Order Line (800) 637-4983
(except in Illinois, Alaska and Hawaii)

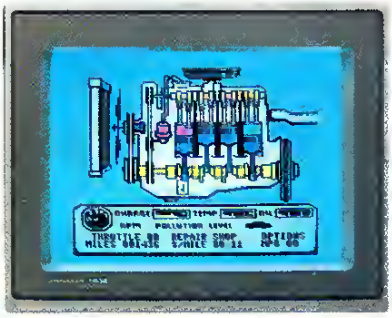
All you need to do this



graph a spreadsheet



write a navel



fix an engine

campase a sang

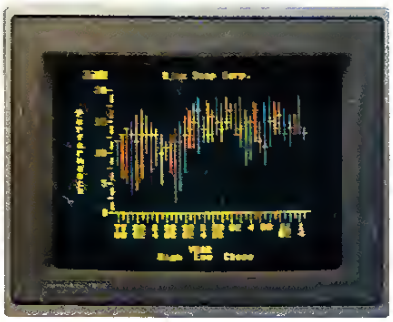
paint a picture

ur banking

learn to fly

organize a data base

tell a stary



farecast sales

When it comes to personal computers, yau want the smartest yau can awn. At a price that makes sense.

The new Cammodare 128™ system has a pawerful 128K memary, expandable to 512K. An 80-column display and 64, 128 and CP/M® modes far easy access ta thausands af educatianal, business and home programs. And a keybaard, with built-in numeric keypad, that aperates with little effort.

Discover the personal computer that does mare far yau. At the price yau've been waiting far. Fram the campany that sells mare persanal computers than IBM® ar Apple.®

COMMODORE 128 PERSONAL COMPUTER

A Higher Intelligence

RETROMAGS

Our goal is to preserve classic video game magazines so that they are not lost permanently.

People interested in helping out in any capacity, please visit us at www.retromags.com.

No profit is made from these scans, nor do we offer anything available from the publishers themselves.

If you come across anyone selling releases from this site, please do not support them and do let us know.

Thank you!